TAMPEREEN TEKNILLINEN YLIOPISTO
TAMPERE UNIVERSITY OF TECHNOLOGY

Matti Pellikka

Finite Element Method for Electromagnetics on Riemannian Manifolds

Topology and Differential Geometry Toolkit

Julkaisu 1209 • Publication 1209

Tampere 2014

Tampereen teknillinen yliopisto. Julkaisu 1209
Tampere University of Technology. Publication 1209

Matti Pellikka

Finite Element Method for Electromagnetics on Riemannian Manifolds

Topology and Differential Geometry Toolkit

Thesis for the degree of Doctor of Science in Technology to be presented with due permission for public examination and criticism in Rakennustalo Building, Auditorium RG202, at Tampere University of Technology, on the 2nd of May 2014, at 12 noon.

Tampereen teknillinen yliopisto - Tampere University of Technology
Tampere 2014

ISBN 978-952-15-3279-5 (printed)
ISBN 978-952-15-3285-6 (PDF)
ISSN 1459-2045

Abstract

This thesis applies new branches of mathematics in computational electromagnetics software. Namely, we consider the application of algebraic topology and differential geometry in finite element modeling. We conclude that from this approach, one can draw benefits to practical electromagnetic modeling. For example, more efficient numerical formulations, field-circuit coupling, and metric and coordinate free modeling techniques.

We present efficient methods for homology and cohomology computation of finite element meshes together with their software implementation. The presented homology and cohomology solver is a part of finite element mesh generator *Gmsh*. Therefore, its use can be easily incorporated into finite element modeling workflow.

We demonstrate the use of homology and cohomology computation results in static and quasistatic electromagnetic field problems. We describe finite element formulations which can be used in lumped parameter extraction from field problems and which can be naturally coupled to electronic circuit problems. Importantly, cohomology computation enables the use of magnetic scalar potential in eddy current problems without any topological restrictions, leading to more efficient and robust field computations.

Lastly, we present a finite element programming environment, where the language of differential geometry has the main role. We interpret the finite element model as a Riemannian manifold, and the fields of interest as differential forms. Using the environment, one can give the computational instructions in metric and coordinate free manner, as the used metric and coordinate system are provided separately. Then, the environment translates the instructions to the actual floating-point operations, which ultimately depend on the used metric and coordinate system. The programming environment implementation builds on top of the *Gmsh* API. That is, we implement tools from differential geometry which utilize an existing finite element framework.

The main contribution of this thesis is the development of these tools to the point where they can be readily expoited in computationally demanding engineering problems. Also, this thesis offers a unified exposition of the needed mathematical concepts and their relation to the electromagnetic field problem formulations.

Preface

I have made this thesis in the inspirational environment of the electromagnetics research group in the Tampere University of Technology. The thesis topic builds on top of the long-term basic research conducted in our research group, to bring some of its results closer to engineering practice.

I have done this work under the supervision of Professor Lauri Kettunen, whose enthusiasm, sincerity, and farsightedness reflects in the atmosphere of the whole research group, creating a close-knit research community. In addition to Lauri Kettunen, I have been privileged to be guided by two other members of our research group, whose advice and perspective complement each other: University Lecturer Saku Suuriniemi and Senior Research Fellow Timo Tarhasaari. Much of my progress is also due to Professor Christophe Geuzaine from the University of Liège, the main developer of the finite element library *Gmsh*. In addition to his encouragement, he enabled and supported the implementation of the methods of this thesis to *Gmsh*.

I would also like to than Lasse Söderlund and Maija-Liisa Paasonen for taking care of administrative tasks, as well as Juha Tampio, Antti Stenvall, Valtteri Lahtinen, Erkki Härö, Arto Poutala, and Olli Pekkola for giving feedback to some of the software implementations I have done for this thesis.

Other people I'd like to thank for more or less casual interaction during my thesis work include Jukka-Pekka Uusitalo, Teemu Rovio, Pasi Raumonen, Janne Keränen, Tuukka Nieminen, Arttu Rasku, Aki Korpela, Risto Mikkonen, Stefan Kurz, and other present and past personnel of our research group who are responsible for its friendly atmosphere.

Lastly, I thank my dear wife Kiti for taking care of the fundamentals on the home front, as well as my son Eevertti for demanding my undivided attention on regular basis; and my daughter Muusa for inspiring me while finalizing this thesis.

Contents

List of symbols

Chapter 2

(E^k, ϕ)	Euclidean reference k-cell, page 37
$(M, \mathcal{K}), (S, \mathcal{L})$	A regular CW-complex of a manifold M or a submanifold S, page 36
(U, x)	A chart, page 31
(V, o)	Oriented vector space, page 30
$[g]$	The equivalence class of g, page 20
$\Lambda^k(T^*M)$	The k-cotangent bundle of a manifold M, page 44
$\Lambda^k(V^*)$	The vector space of alternating k-linear maps on a vector space V, page 28
$\mathcal{D}$	A differentiable atlas, page 31
∂M	The boundary of a manifold, page 32
∂, ∂_k	(k:th) boundary homomorphism, page 21
β_k	The k:th Betti number, page 27
w^k	A Whitney k-form, page 49
Σ^k	A basis of a k-cochain group, page 40
Σ_k	A basis of a k-chain group, page 39
δ, δ_k	(k:th) coboundary homomorphism, page 23
$\mathcal{C}, \mathcal{C}', \check{\mathcal{C}}, \bar{\mathcal{C}}$	Chain complex, page 21
$.^H$	The conjugate/Hermitian transpose of a column vector or a matrix, page 26
$.^T$	The transpose of a column vector or a matrix, page 19
σ^k	A k-cell, page 35
ϵ_k	A basis k-cochain, page 40
$\mathbb{C}$	The field of complex numbers, page 32
$\mathrm{d}x^i$	i:th coordinate basis covector, page 33
d	The exterior derivative of a differential form, page 44
δ^I_J	Kronecker delta for multi-indices, page 28

Δ^k	Standard k-simplex, page 37
$\det \mathrm{A}$	The determinant of a matrix A, page 28
$\dim G$	Rank of the free subgroup of the finitely generated free abelian group G, page 20
$\dim V$	The dimension of a vector space V, page 25
$\mathbb{E}^n$	The n-dimensional Euclidean space, page 29
γ, γ_k	a k-cochain, page 23
$\mathcal{H}^k(M, S)$	The space of harmonic differential k-forms, page 47
$\operatorname{im} f$	Image of a map f, page 18
$\iota \cdot\cdot$	The incidence relation of cells, page 36
$\langle \cdot, \cdot \rangle$	An inner product, page 26
$\mathbb{Z}$	The ring of integers, page 19
$\mathbb{Z}^n$	The group of n-tuples of integers, page 19
$\mathbb{Z}_n$	The group of integers modulo n, page 20
$\ker f$	Kernel of a map f, page 18
λ_i	A finite element shape function at i:th node, page 37
D_k	The matrix representation of the k:th boundary homomorphism, page 39
G	The coordinate chart representation of a metric tensor g, page 34
$\| \cdot \|$	A norm, page 30
ω	An element of a dual space of a vector space, page 26
$\omega^\sharp$	The sharp operator, page 26
$\Omega^k(M), \Omega^k(M, S)$	The vector space of differential k-forms on a manifold M (whose trace vanish on a submanifold S), page 44
$\frac{\partial}{\partial x^i}$	i:th coordinate basis vector, page 32
$\mathbb{R}$	The field of real numbers, page 29
$\mathbb{R}^n_+$	The half-space of the real coordinate space $\mathbb{R}^n$, page 30
$\mathbb{F}$	Field of real or complex numbers, page 25
$\mathbb{F}^n$	Vector space of n-tuples of real or complex numbers, page 25
$\sim$	Equivalence relation, page 20
$\star$	The Hodge operator, page 29
t, t_S	The trace operator of a differential form (to a submanifold S, page 44
τ	A topology, page 30
$\bigotimes^{(}_{k,l}(T)M)$	The (k, l)-tensor bundle of a manifold M, page 109

$\boldsymbol{\omega}$	The coefficient representation of ω, page 26
$\mathbf{v}$	The coefficient representation of v, page 25
$\wedge$	The exterior product, page 28
ζ, ζ_k	a k-cocycle, page 24
$B^k(\mathcal{C}; G)$	k:th coboundary group, page 24
B_k	k:th boundary group, page 21
$B_r^n(y)$	Open y-centered r-radius Euclidean n-ball, page 30
c, c_k	a k-chain, page 21
$C^k(\mathcal{C}; G)$	k:th cochain group of a chain complex $\mathcal{C}$, page 23
$C^k(M), C_k(M, S)$	The k:th cochain group of a manifold M (relative to a submanifold S), page 40
C_k	k:th chain group, page 21
$C_k(M), C_k(M, S)$	The k:th chain group of a manifold M (relative to a submanifold S), page 38
F	A Free subgroup, page 20
g	The metric tensor, page 34
G/H	The quotient group of groups G and H, page 20
$H\Omega^k(M, S)$	The Sobolev space of differential k-forms, page 47
$H^k(\mathcal{C}; G)$	k:th cohomology group of the chain complex $\mathcal{C}$, page 24
$H^k(M; G), H_k(M, S; G)$	The k:th cohomology group of a manifold M (relative to a submanifold S), page 41
$H^k_{\mathrm{dR}}(M, S)$	Relative k:th de Rham cohomology space of a manifold M and a submanifold S, page 46
$H_k(\mathcal{C})$	k:th homology group of the chain complex $\mathcal{C}$, page 21
$H_k(M), H_k(M, S)$	The k:th homology group of a manifold M (relative to a submanifold S), page 41
I, J	Multi-index, page 28
K, L	A cell complex, page 36
M	A differentiable manifold, page 31
T	The torsion subgroup, page 20
T_pM	A tangent space of a manifold M at p, page 33
T_p^*M	A cotangent space of a manifold M at p, page 33
TM	The tangent bundle of a manifold M, page 43
v	An element of a vector space, page 25
V^*	The dual space of a vector space V, page 26
$v^\flat$	The flat operator, page 26

$W^k(M, S)$	The vector space of Whitney k-forms, page 49
z, z_k	a k-cycle, page 21
$Z^k(\mathcal{C}; G)$	k:th cocycle group, page 24
Z_k	k:th cycle group, page 21
$\binom{n}{k}$	The binomial coefficient, page 28

Chapter 3

(b, a)	A reduction pair, page 55
H^k	The basis representation matrix of a k:th cohomology group, page 63
H_k	The basis representation matrix of a k:th homology group, page 63
$O(\cdot)$	The computational complexity class, page 57
Q	Queue data structure, page 59
$R(\cdot, \cdot)$	The computational complexity of the `removeReductionPair` routine, page 57

Chapter 4

∂_t	The partial derivative with respect to time, page 80
E	A 1-cohomology basis function, page 83
e	A Whitney 1-form, page 83
F	A 2-cohomology basis function, page 83
f	A Whitney 2-form, page 83
N	A 0-cohomology basis function, page 83
n	A Whitney 0-form, page 83
W_k	A k-cohomology basis function, page 83
ϵ	The permittivity (1,1)-tensor, page 80
μ	The permeability (1,1)-tensor, page 80
Ω	A 3-submanifold of $\mathbb{R}^3$, page 80
ρ	The charge density 3-form, page 80
Σ	A 2-submanifold of $\mathbb{R}^3$, page 80
σ	The conductivity (1,1)-tensor, page 80
b	The magnetic flux density 2-form, page 80
d	The electric flux density 2-form, page 80
e	The electric field 1-form, page 80
h	The magnetic field 1-form, page 80
j	The current density 2-form, page 80

$\mathrm{J}_M(\mathbf{p})$	A Jacobian matrix of the map ϕ_M at a point $\mathbf{p}$, page 105
μ	A map from preprocessor model to a manifold M, page 104
ϕ_σ	A map from euclidean reference cell of a finite element to the preprocessor model, page 106
ϕ_M	A map from preprocessor model to a coordinate chart of a manifold M, page 104
φ	A map from m-manifold M to n-manifold N, page 104
$\mathbf{p}$	Model coordinates of a point p on a manifold M, page 107
$\mathbf{u}$	Finite element coordinates of a point p on a manifold M, page 106
$\mathbf{x}$	Manifold coordinates of a point p on a manifold M, page 105
M	An m-manifold, page 104
N	An n-manifold, page 104

Chapter 1

Introduction

The finite element method is a widely applied method in sciences and engineering to solve *boundary value problems*. A boundary value problem has three ingredients

1. A *regular domain*[1] and its parametrization.
2. Equations of unknown functions and their partial derivatives, i.e. *partial differential equations*, that are required to hold in the domain.
3. Equations of unknown functions and their partial derivatives that are required to hold on the boundary of the domain, i.e. *boundary conditions*.

In the finite element method, the numerical solution of the boundary value problem stands on two concepts. The decomposition of a domain to *finite elements* to approximate the unknown field, and the evaluation of *bilinear functionals* associated with the finite elements in the domain and on its boundary. Traditionally, the domain is interpreted as a subset of the *Euclidean space*, while the bilinear functionals arise from the integration of a dot product of vector fields or a product of scalar fields.

In this thesis, we take an axiomatic approach to the mathematical structures utilized in the boundary value problems and reinterpret the finite element method accordingly. The cleanest means to achieve this would be to utilize the *category theory* [25], instead of the set theory. However, since the main audience of this thesis is engineering practitioners, rather than mathematicians, we try to limit concepts not included in the usual finite element modeling curriculum.

The reinterpretation of the finite element method involves new computational steps and software design aspects to accommodate to the changes. The reinterpretation has two main themes, the global topology and the metric of the domain.

The domain of a boundary value problem can have topological features that are considered to be global in contrast to local. Such features are the existence of holes of different dimension in the domain, such as tunnels or voids. It turns out, that these features together with the boundary conditions of the unknown field need to be taken into account to ensure the existence and the uniqueness of the solution of a boundary value problem. In other words, such analysis is an indispensable part of the boundary value problems.

[1] A *weakly Lipschitz domain* [27]

To this end, we have studied computational methods to detect such holes their relation to the boundary conditions, and means to take them into account in the finite element method. Such study is called computational homology and cohomology. In the finite element setting, the starting point of the homology and the cohomology computation is the finite element mesh together with the information where the unknown field is constrained by the boundary conditions.

Homology and cohomology has the most visible appearance the in the electromagnetic modeling. Already the Maxwell's equations hint to this direction, if one studies in their integral form the interplay between the integration domain and its boundary. Historically, the conscious interest in the homology computation in electromagnetic modeling arose from the need to compute "cuts" for the magnetic scalar potential in domains with tunnels. However, as we argue in this thesis, cohomology permeates all electromagnetic boundary value problems and cohomology computation is a powerful tool in vector potential formulations and in the field-circuit coupling in electromagnetic modeling. To this end, we present finite element formulations of electromagnetic field problems that exploit the results of cohomology computation.

The first aim of this thesis is to make cohomology computation a commonplace practice in the finite element modeling. Therefore, in the course of this thesis we have implemented a homology and cohomology solver as an integrated part of a finite element mesh generator.

By metric we refer to local measure of distances and angles within the domain. In the framework of the Euclidean space, it is assumed that the metric is constant with respect to Cartesian coordinates across the domain. Such metric is called *Euclidean metric*. In the generalization called *Riemannian manifold*, metric can vary from point to point, stripping the metric meaning from the coordinate numbers. The object that carries the information of the local metric is called the *metric tensor*.

With the Euclidean space interpretation of the domain the classical vector analysis formalism is often used, which has three shortcomings. First, the formalism is hard-wired to the Euclidean metric, and special care is needed to use a general metric tensor. Second, the expressions often depend on the employed coordinate system of the domain. This needlessly obscures the notation with excessive details. Third, classical vector analysis is devised for 3-dimensional domains, which makes the notation cumbersome in 2-dimensional domains and the formalism insufficient for higher dimensional domains. Therefore, we have adapted the formalism of *differential geometry* in this thesis.

The Riemannian manifold interpretation of the domain together with the differential geometry formalism are the ingredients of the second major topic of the thesis. We have implemented a finite element programming environment, which performs the Riemannian manifold interpretation and allows the user to program finite element solvers using the language of the differential geometry.

The framework of the differential geometry overcomes the mentioned limitations of the classical vector analysis. It is indifferent about the used metric, the coordinate system, and the dimension of the domain. That is, expressions are invariant under such changes. In a computational setting, such feature can easily be exploited with an interface where the metric and coordinate system are provided separately from the expressions encoding computational procedures. The software then translates the expressions to the actual numerical computations that ultimately depend on the used metric and coordinate system.

That is, one is able to write programs in a way that resembles the workflow of an applied mathematician. This constitutes the second aim of this thesis.

1.1 Background, motivation, and usefulness of the research

The interest in homology and differential geometry in computational electromagnetics research community rose in the 1980s. The impetus for this was the increased performance of computers that made the numerical solution of 3-dimensional field problems possible in acceptable time. It was soon noticed however, that the application of the scalar and vector potential formulations to three dimensions was not straightforward.

1.1.1 Homology and cohomology computation

The magnetic scalar potential formulation of magnetostatics requires that there's a "cut" in the domain where the magnetic scalar potential is multi-valued. The jump of the scalar potential across the cut equals the magnetomotive force that generates the magnetic field. In two-dimensional problems, the cut was easy to designate manually. However in a three-dimensional field problem, the room for complexity of the modeling domain is much larger. This initiated a research for algorithms to produce the cuts automatically, even without having an exact definition of the "cut" in mind. Even today, these ad hoc algorithms are widely used in commercial simulation software.

It was discovered that the exact definition of the cut involves algebraic topology concept called homology[2]. An algorithm for the cut computation in finite element meshes was introduced in [45]. However, the inexact algorithms that worked in many practical situations persisted commercially.

At the time, the general homology computation was believed to be computationally expensive as it involves the computation of *Smith normal form* integer matrix decomposition. For this reason, general homology computation was not pursued in computational electromagnetics. However, the research on general homology computation continued on a path where the size of the homological problem was first made smaller with so-called "reduction techniques" [39], which have acceptable efficiency.

Also an another approach was considered. Instead of producing a cut for the scalar potential, one can produce a "source field" or "loop field" to accompany the scalar potential. Such vector field belongs to the 1-cohomology class, and specialized algorithms based on spanning trees and 1-homology computation were developed for 1-cohomology computation. Such approach is more convenient than cut computation in magnetoquasistatic problems where both scalar and vector potentials are needed.

In the three dimensional vector potential formulations, it was noticed that the net quantities such as the net current or the magnetic flux are troublesome to impose using the boundary conditions of the vector potential. A completely non-intuitive 1-cohomology

[2]The cut in a domain $M \subset \mathbb{R}^3$ is a representative of an element of the relative 2-homology group $H_2(M, \partial M)$. The representative is called a **cut** if it is an orientable submanifold of M and can be embedded in $\mathbb{R}^3$ [43].

vector field needs to be constructed on the boundary. The robust technique to construct such vector fields involves cohomology computation on the boundary of the domain.

In the both above examples, the homology or cohomology relates to the net quantities. Net quantities are the state-variables in the circuit models of the electromagnetism. Therefore, it should be just a matter of formulation to exploit cohomology computation in the field-circuit coupling in electrical engineering.

The evident topological problems in both scalar and vector potential approaches and the relation to the circuit coupling motivated the research done for this thesis. To prepare for unforeseen applications, we solve for general, possibly relative, homology and cohomology groups, not just for 1-homology and 1-cohomology that are known to be important in the applications of the computational electromagnetics.

As an outcome, we now have an efficient homology and cohomology solver within a finite element mesh generator. Therefore, it can be easily integrated to the usual finite element modeling workflow. Furthermore, we have demonstrated its usage in industrial scale electromagnetic field problems which shows that it is a viable tool in electrical engineering.

1.1.2 Differential geometry and Riemannian manifolds

Soon after the introduction of the vector potentials to the finite element method it was discovered that the mere "three-component scalar potential" is not the right approach for vectorial finite elements in electromagnetism[7]. Such approximation enforces the continuity of all three components across the finite element interfaces, while only tangential or normal continuity conforms to the boundary conditions derived from the Maxwell's equations.

As a solution, so-called edge and facet elements, Whitney forms in general, soon penetrated the research community and commercial simulation software. While the Whitney forms are *differential forms* living on a differentiable manifold, the viewpoint that they are vector fields in the Euclidean space still persists. However, the interest what else can be gained from the framework of differential geometry in practical electrical engineering was sparked.

The differential form interpretation fits naturally to the geometric finite element setting and makes it possible to develop software the separates metric and coordinates from expressions that are used to compute the elements of the final system matrix.

Using other than the Euclidean metric in the computations has been found beneficial in many modeling techniques. Some of them are called "transformation techniques" [34], where the modeling domain is still the Euclidean space, but one can perform the computations on user-defined non-Cartesian coordinate chart instead of the Cartesian one. In a more general setting of the Riemannian manifold one can consider domains for which no Cartesian coordinate chart exist. For example, one can perform computations on a curved surface using a flat finite element mesh. Both of these techniques are covered by the Riemannian manifold interpretation of the domain of a boundary value problem.

However, the Riemannian manifold interpretation increases the complexity of a finite element solver, since the free choices of the coordinate chart and the metric need to be taken into account in the numerical computations. The complexity can be alleviated by the inclusion of an additional layer to the software on which computational instruc-

tions can be written in terms of the differential geometry. Underneath this layer, the software translates the expressions into coordinate chart and metric dependent floating-point operations. Such programming interface has been developed on the course of this thesis.

1.2 Survey of recent research

The role of homology computation in computational electromagnetics was first recognized in [42] and later considered in [43, 44, 6, 45]. However, efficient general homology computation methods we're not available to fully exploit the findings at the time.

First steps towards efficient general homology computation were the introduction of *reduction of chain complexes* [39, 38] in computational homology. Similar algorithms, but based on homotopy invariance of homology rather than properties of chain complexes were discussed in context of finite element meshes and electromagnetics in [61].

Since in the computational electromagnetism 1-cohomology computation has the most importance, specialized algorithms for 1-cohomology computation based on 1-homology computation are considered in [21, 22]. The results of such computation are often called *thick-cuts*, recognized in [41]. Direct 1-cohomology computation methods for computational electromagnetics were considered in [18, 19].

The road to general cohomology computation by the reduction of chain complexes was paved by the *Coreduction homology algorithm* [47], which was then applied to cohomology computation of finite element meshes in [49] by the author.

Other software that perform homology and cohomology computation include *CHomP* [12, 38], *javaPlex* [62, 11], and *GAP homology* [37, 36] packages. The design objectives of these popular packages are slightly different from ours, with less emphasis on problem position in finite element modeling.

Earlier, a Riemannian manifold approach to the finite element modeling has been taken in *Manifold Code* [35], as it separates the metric from the coordinate representation of the domain. However, the metric tensor is defined indirectly by providing the weak forms in contrast to our approach. It aims to solve "second-order nonlinear elliptic systems of partial differential equations on domains with the structure of Riemannian two- and three-manifolds".

A similar project called *PyDEC* [4, 5] is a Python library for finite element and discrete exterior calculus. In *PyDEC*, the metric tensor is the pullback metric of an embedding of the simplicial complex into an n-dimensional Euclidean space.

1.3 Original contributions

In this section we specify the original contributions by the author that to our knowledge first appear in this thesis or in an earlier article by the author.

1.3.1 Development of reduction techniques for homology and cohomology computation

The reduction techniques for the homology computation considered in [61] had some shortcomings. They are inefficient in the computation of absolute homology of closed manifolds and in the computation of relative homology where the boundary of the manifold is the relative subdomain.

In chapter 3 of this thesis and in the article [49], the author applies *chain equivalences* inspired by the Coreduction Homology Algorithm [47] in homology computation. The main difference in our approach with respect to the Coreduction Homology Algorithm is that the author is able to also produce the representatives of the homology group generators.

The algorithms considered in [61] and the above homology algorithm are also applied by the author in the cohomology computation to produce the representatives of the cohomology group generators.

1.3.2 Implementation of homology and cohomology solver

On the course of this thesis, the author has implemented a homology and cohomology solver as an integrated part of the finite element mesh generator *Gmsh*. It is described in the chapter 3 and appears in the articles [49], [49],[50], and [51].

The specification of the input and output, the user interface, the reduction techniques, and the homology and cohomology computation steps are implemented by the author. For the Smith normal form computation, a subroutine within the homology and cohomology computation steps, a library developed by the author of [61] was used.

1.3.3 Cohomology based formulations of the electromagnetic boundary value problems

The cohomology has appeared in a form or another in the electromagnetic boundary value problem formulations for a long time. What has been lacking however, is a general finite element formulation framework where the cohomology has an equipotent role it should have.

In chapter 4 the author represents such framework and coins the term *Cohomology basis function* to bring cohomology in front matter from behind the scenes. In the article [51] the author applies the framework to a magnetic field oriented formulation of a magnetoquasistatic problem.

The formulation framework makes it easier to recognize which parts of the problem stem from cohomology. For instance, it makes clear that to assign electric potential values on equipotential boundary surfaces, a common task in everyday electromagnetic modeling, is the same thing as choosing the cohomology class of the solution.

The formulation framework unifies the aspects of many formulations. For example "floating potentials" [17], "thick-cuts" or "source fields" [41, 55, 33], and "thick-links" or "thinned conductors" [60, 20], and it clarifies the duality [16] of complementary formulations. In the framework of the author, all these terms are just different instances of the cohomology basis functions at work.

1.3.4 Implementation of Riemannian manifold programming interface

The author has designed and implemented a `C++` programming interface that imitates the structure and objects defined on the Riemannian manifold. The interface utilizes the application programming interface of *Gmsh* to provide the usual finite element data structures. The interface is described in chapter 5 and in the article [52].

The main benefit of the interface is that it allows one to write programs using the language of the differential geometry, which is inherently coordinate and metric free. However, the description of a finite element mesh and the actual computations in the finite element method depend on the coordinates and the metric. This gap has been filled by the author in the represented library.

1.4 Organization

The organization of this thesis is as follows. In chapter 2 we define the mathematical concepts employed in this thesis. We begin with purely algebraic concepts and the concept of manifold which is used to model the domain of a boundary value problem. Then, these two tracks are merged to define homology and cohomology of a manifold, and to define differential forms that model the fields on a manifold. Although this is chapter is standard material in many textbooks, we find it necessary to be able to read this thesis as an independent piece of work.

In chapter 3 we describe how the homology and the cohomology of a manifold is computed. The computation has three stages: the construction of the chain complex, the exploitation of the reduction techniques on the chain complex, and finally the solution of the algebraic problems using integer arithmetic. In the chapter, we also consider post-processing techniques of the homology and the cohomology computation results make them more appealing for visualization in exploitation in the solution of boundary value problems. In this chapter we combine the previous research to our original contributions on the subject.

In chapter 4 we formulate static and magnetoquasistatic boundary value problems that exploit the results of the homology and the cohomology computation. The chapter also clarifies the engineering significance of homology and cohomology in electromagnetics. While the ingredients of this chapter can be found from earlier computational electromagnetics research, it provides a novel, unified framework of cohomology basis functions in the formulation of boundary value problems.

In chapter 5 we present a Riemannian manifold programming interface in a finite element environment. We describe how the finite element preprocessor model is interpreted as a Riemannian manifold, and how tensorial objects on the manifold are imitated within a computational environment. We also provide the motivation for such interface and examples of its application.

Chapter 2

Mathematical concepts

In this chapter, we introduce the mathematical structures and theorems that are discussed and utilized in this thesis. While most of the contents of this chapter can be found from textbooks [29, 25, 8, 63, 23, 24, 48, 31, 57], we find it necessary to condense the topics that are relevant to this thesis in a notionally uniform exposition. We also want to emphasize the structured and minimalist viewpoint of mathematics and mathematical modeling.

We also pay close attention to the *representations* of the elements of mathematical objects. In computations, one is ultimately dependent on integer and floating-point number arithmetic. Thus in order to imitate elements of mathematical structures in computations and in software, one needs to represent them as n-tuples of floating-point numbers or integers. Such representations are not unique, and turning this non-uniqueness into an advantage is a recurring topic in this thesis.

In engineering, one utilizes a myriad of mathematical structures and theorems to solve a problem. Often, the structures are not identified, but treated as a single, inseparable mix of mathematics. In this thesis, we do the opposite. While we cannot escape the fact that for example the solution of a boundary value problem utilizes a variety of mathematical ideas, we can sort out some of the ingredients of the mix. The benefit of doing so is that once we have identified the structures at play, we can utilize the mathematical theorems regarding that single structure in engineering. That is, recognizing whether a mathematical theorem is useful in an engineering problem becomes a manageable task.

This "divide and conquer" of mathematical structures is used to establish the usefulness of the following theorems of modern mathematics in the electromagnetic modeling: *Lefschetz duality theorem*, *de Rham's theorem*, and *Hodge's theorem*. The way the theorems are stated in the provided original and secondary sources is intangible to a working engineer. In this thesis, we hope to narrow the gap.

In physics, the term *general covariance* is used to mean that one can use any coordinate system one chooses to model physical phenomena. Such property is the foundation, rather than a theorem, in the theory of *Riemannian manifolds*. Therefore, such mathematical formalism should also empower engineering practice, which is why this thesis is written within that framework.

2.1 Algebraic structures

Algebraic structures lay foundations for the applied mathematics. Most of the engineering practice revolves around linear algebra which is the study of an algebraic structure called the *vector space*. Very often engineering problems are turned into problems of linear algebra, for which efficient computational algorithms do exist. However, in this thesis we encounter problems that are more naturally expressed in terms of less familiar algebraic structures.

Algebraic structures apply the idea of a *closure system* repeatedly. Consider a set with a binary operation for the elements of the set. If the result of the binary operation for all arguments belongs to the set, the set is called *closed* under that binary operation. Another important concept is a structure-preserving map between sets, called *homomorphism*. It is a generalization of the *linear map* between vector spaces. That is, it makes no difference whether one applies a binary operation before or after the mapping.

In this thesis, we encounter algebraic structures that are closely related to the vector spaces and apply them to engineering problems. As the mere linear algebra formalism does not suit all of our needs, we need to expand our view to the close relatives of the vector space: *abelian groups* and *graded* vector spaces called *exterior algebra*.

2.1.1 Abelian group

Abelian group is an algebraic structure that underlies in many mathematical structures like *vector space* and *field*. It also has significance of its own right. For example in this thesis, *homology* and *cohomology* are first defined in terms of the abelian groups alone.

Let G be a *set* and let $+ : G \times G \to G$ be a *commutative* and *associative* map [25]. Then, the pair $(G, +)$ is an **abelian group** if there exists an element $0 \in G$ such that $a+0 = a$ holds and if there exists an element $-a \in G$ for all $a \in G$ for which $a+(-a) = 0$ holds. A pair $(A, +)$ is a **subgroup** of $(G, +)$, if $A \subset G$ holds and the pair $(A, +)$ is a group of its own right.

A map $f : G \to H$ from a group $(G, +)$ to a group $(H, \oplus)$ is a **group homomorphism** if $f(a+b) = f(a) \oplus f(b)$ holds for all $a, b \in G$. If a group homomorphism is *injective* and *surjective*, it is a **group isomorphism**. When the context permits, an abelian group $(G, +)$ is denoted with plain G. The **kernel** and **image** of a homomorphism f are sets

$$\ker f := \{g \in G \,|\, f(g) = 0\} \quad \text{and} \tag{2.1}$$

$$\operatorname{im} f := \{h \in H \,|\, h = f(g), g \in G\}, \tag{2.2}$$

which turn out to be subgroups of G and H, respectively [25].

The operations $a + (-b)$, $\sum_k a = a + a + \ldots + a$, and $\sum_k -a = -a + (-a) + \ldots + (-a)$ are abbreviated as $a - b$, ka, and $-ka$, respectively. With these conventions, the algebraic structures of the abelian group and the *module over the ring of integers* are indistinguishable. Such structure is closely related to the vector space, with the difference that the coefficients are integers rather than real numbers.

A subset $S \subset G$, $0 \notin S$, is a **generating set** of the group G if any element g of the set G can be expressed as a finite composition of the elements of S and their inverses

with the map +:

$$g = k^1 s_1 + k^2 s_2 + \ldots + k^n s_k = \sum_{i=1}^{n} k^i s_i, \quad \forall\, i: \quad k^i \in \mathbb{Z}, \quad s_i \in S. \tag{2.3}$$

If there exists such generating set that such composition is unique, G is a **free abelian group**. If the set S is finite, G is **finitely generated abelian group**. Such generating set of a finitely generated free abelian group is called a **basis**. The number of basis elements is called the **rank** of the free abelian group.

Formal sum

From ***any*** finite set S one can construct a finitely generated abelian group G. One declares that the elements g of the group are **formal sums** of the elements of the set S:

$$g := \sum_{i=1}^{n} k^i s_i, \quad \forall\, i: \quad k^i \in \mathbb{Z},\, s_i \in S, \tag{2.4}$$

where $n = |S|$ is the cardinality of S and $s_i \neq s_j$ when $i \neq j$. The map + and the inverse element are defined by

$$g_1 + g_2 = \sum_{i=1}^{n} k^i s_i + \sum_{i=1}^{n} q^i s_i := \sum_{i=1}^{n} (k^i + q^i) s_i. \tag{2.5}$$

$$-g = -\Big(\sum_{i=1}^{n} k^i s_i \Big) := \sum_{i=1}^{n} (-k^i) s_i. \tag{2.6}$$

The set S is a basis for such group, and an element of such group can be *represented* in that basis as a integer vector $\mathbf{k} = \begin{bmatrix} k^1 & k^2 & \ldots k^n \end{bmatrix}^T \in \mathbb{Z}^n$ of the coefficients k^i in the formal sum. Then, a group element $g \in G$ is denoted by $g = (s^1, s^2, \ldots, s^n)\mathbf{k} = \mathrm{S}\mathbf{k}$.

The integer vectors $\mathbf{k}$ themselves constitute a finitely generated free abelian group $\mathbb{Z}^n$ under the vector addition. This group is *isomorphic* to the abelian group G constructed via the formal sum. More generally, any finitely generated free abelian group is isomorphic to the group $\mathbb{Z}^n$ [48].

Let H be a finitely generated free abelian group with a basis $R = (r^1, r^2, \ldots, r^m)$. Then, a group homomorphism $\phi : G \to H$ satisfies $\phi(s_i) = (r^1, r^2, \ldots, r^m)\boldsymbol{\phi}_i$ for some $\boldsymbol{\phi}_i \in \mathbb{Z}^m$. That is, one can construct a $m \times n$ matrix $\mathrm{F} = \begin{bmatrix} \boldsymbol{\phi}_1 & \boldsymbol{\phi}_2 & \ldots \boldsymbol{\phi}_n \end{bmatrix}$ that represents the homomorphism ϕ. These observations enable one to solve algebraic problems associated with the "abstract" groups G and H by computations that involve the "concrete" integer vector groups $\mathbb{Z}^n$ and $\mathbb{Z}^m$. In other words, the following diagram of finitely generated free abelian groups *commutes*:

$$\begin{array}{ccc} G & \xrightarrow{\ \phi\ } & H \\ \updownarrow & & \updownarrow \\ \mathbb{Z}^n & \xrightarrow{\ \mathrm{F}\ } & \mathbb{Z}^m \end{array} \tag{2.7}$$

Torsion subgroups

If an abelian group is *not* free, it has a non-trivial *torsion subgroup*. An element $g \in G$ is a **torsion element** if $\sum_t g = 0$ holds for some non-zero integer t, called the **torsion coefficient** of g. Each torsion element generates a torsion **torsion subgroup** T of G. That is, the elements u of a torsion subgroup generated by g are of the form

$$u = qg = \sum_q g, \quad q \in \mathbb{Z}_t, \tag{2.8}$$

where $\mathbb{Z}_t$ is the *group of integers modulo* t. Therefore, a torsion subgroup with the torsion coefficient t is isomorphic to the group $\mathbb{Z}_t$. That is, an integer $q \in \mathbb{Z}_t$ is a representation for u.

Characterization of finitely generated abelian groups

Let G be a finitely generated abelian group with a minimal generating set $S = \{g_1, g_2, \ldots, g_n\}$. Then, an element $g \in G$ can be written as a *unique* sum [48]

$$g = \sum_{i=1}^{r} k^i g_i + \sum_{j=r+1}^{n} q^{j-r} g_j, \quad k^i \in \mathbb{Z}, \quad q^j \in \mathbb{Z}_{t_j} \tag{2.9}$$

The element $g \in G$ is then represented by a tuple $((k^1, k^2, \ldots, k^r), q^1, \ldots q^{n-r}) \in \mathbb{Z}^r \times \mathbb{Z}_{t_1} \times \ldots \times \mathbb{Z}_{t_{n-r}}$ with respect to the generating set S. It follows that G is isomorphic to the group formed by the *internal direct sum* $\mathbb{Z}^r \oplus \mathbb{Z}_{t_1} \oplus \ldots \oplus \mathbb{Z}_{t_{n-r}}$. The group G has thus the decomposition $G = F \oplus T$, where F is a rank r free subgroup and T is the internal direct sum of torsion subgroups of G with torsion coefficients t_j. The rank r and the torsion coefficients t_j are uniquely determined by G, even though the generating set S is not unique [48]. By $\dim G$ we denote the rank of the free subgroup of the finitely generated free abelian group G.

Quotient group and the short exact sequence

Let G be an abelian group and H its subgroup. Define an *equivalence relation* $\sim$ in G to be such that

$$g_1 \sim g_2 \iff g_1 - g_2 \in H \tag{2.10}$$

A **coset** $[g]$ of $g \in G$ is the set

$$[g] := \{g + h \,|\, h \in H\} := g + H \tag{2.11}$$

of equivalent elements in G. The **quotient group** $G/H := \{g + H \,|\, g \in G\}$ is an abelian group whose elements are such cosets. The map $+$ and the inverse element are defined by

$$[g_1] + [g_2] := [g_1 + g_2], \tag{2.12}$$

$$-[g] := [-g], \tag{2.13}$$

and since for $h \in H$, $[g] + [h] = [g + h] = [g]$ holds, the coset $[h]$ is the zero element of G/H.

The inclusion map $i : H \to G$, $i(h) = h$, is an injective homomorphism. Define a surjective homomorphism $j : G \to G/H$ by $j(g) = [g]$. Then $j \circ i = 0$ holds since $j(i(h)) = j(h) = [h] = 0$ holds for all $h \in H$. That is, $\operatorname{im} i \subset \ker j$ holds. Moreover, since $g \in H = \operatorname{im} i$ holds for all $g \in \ker j$ we obtain that $\ker j = \operatorname{im} i$ holds. It is said that then the sequence of abelian groups

$$\ldots \to H \xrightarrow{i} G \xrightarrow{j} G/H \to \ldots \tag{2.14}$$

is *exact* at G. Since $\ker i = \{0\}$ and $\operatorname{im} j = G/H$ hold due to injectivity of i and surjectivity of j, the sequence can be completed to form a **short exact sequence**

$$0 \to H \xrightarrow{i} G \xrightarrow{j} G/H \to 0 \tag{2.15}$$

which is exact at H, G, and G/H. By the *first isomorphism theorem* [48], the group G/H is isomorphic to the group $G/\ker j = G/\operatorname{im} i$, where the equality holds by the exactness of the sequence. This indicates that ***the map*** $i : H \to G$ ***alone characterizes the structure of the quotient group*** G/H.

2.1.2 Homological algebra

In homological algebra, one studies the properties of the chain complexes and relations between them. A **chain complex** $\mathcal{C} = (C_\bullet, \partial_\bullet)$ is a sequence of *abelian groups* $C_\bullet$ and *homomorphisms* $\partial_\bullet$ between them, usually presented as a diagram

$$\ldots \xrightarrow{\partial_{k+2}} C_{k+1} \xrightarrow{\partial_{k+1}} C_k \xrightarrow{\partial_k} C_{k-1} \xrightarrow{\partial_{k-1}} \ldots \tag{2.16}$$

The sequence is required to have the complex property that $\partial_k \circ \partial_{k+1} = 0$ holds for every k. Therefore, $\operatorname{im} \partial_{k+1} \subset \ker \partial_k$ holds. However, the sequence is not necessarily *exact*, for an exact sequence $\operatorname{im} \partial_{k+1} = \ker \partial_k$ would always hold.

The homomorphisms ∂_k that satisfy $\partial_k \circ \partial_{k+1} = 0$ are called **boundary homomorphisms**. If the degree k of ∂_k is clear from the context, we denote it by plain ∂. Later in this thesis, the abelian groups $C_\bullet$ are often vector spaces and the homomorphisms $\partial_\bullet$ are then linear maps.

The elements of the subgroups $Z_k = \ker \partial_k$ and $B_k = \operatorname{im} \partial_{k+1}$ of C_k are called k**-cycles** and k**-boundaries**, respectively. The quotient group

$$H_k(\mathcal{C}) := Z_k/B_k = \{z + B_k \,|\, z \in Z_k\} \tag{2.17}$$

is called the k**:th homology group** of the chain complex $\mathcal{C}$. That is, the elements $[z]$ of $H_k(\mathcal{C})$ are equivalence classes of Z_k with the equivalence relation

$$z \simeq z' \iff z = z' + \partial c, \tag{2.18}$$

where $z, z' \in Z_k$ and $c \in C_{k+1}$. The k-cycles z and z' are then called **homologous**. Since B_k is a subgroup of Z_k, the sequence

$$0 \xrightarrow{i} B_k \xrightarrow{j} Z_k \to H_k(\mathcal{C}) \to 0 \tag{2.19}$$

is exact.

Let $\mathcal{C}' = (C'_\bullet, \partial'_\bullet)$ be an another chain complex. The collection of homomorphisms $\phi_k : C_k \to C'_k$ is called a **chain map** if it satisfies $\phi_k \circ \partial_{k+1} = \partial'_{k+1} \circ \phi_{k+1}$ for all k. This property is presented as a diagram by saying that the following diagram *commutes*:

$$\begin{array}{ccccccccc} \cdots & \xrightarrow{\partial_{k+2}} & C_{k+1} & \xrightarrow{\partial_{k+1}} & C_k & \xrightarrow{\partial_k} & \cdots \\ & & \downarrow{\scriptstyle \phi_{k+1}} & & \downarrow{\scriptstyle \phi_k} & & \\ \cdots & \xrightarrow[\partial'_{k+2}]{} & C'_{k+1} & \xrightarrow[\partial'_{k+1}]{} & C'_k & \xrightarrow[\partial'_k]{} & \cdots \end{array} \tag{2.20}$$

If the degree k of a chain map ϕ_k is clear from the context, we denote it by plain ϕ.

Chain maps

Chain maps are important since they induce homomorphisms $\phi_* : H_k(\mathcal{C}) \to H_k(\mathcal{C}')$ between the homology groups of the chain complexes $\mathcal{C}$ and $\mathcal{C}'$. The induced map is defined by

$$\phi_*([z]) := [\phi(z)] \quad \forall\, z \in Z_k. \tag{2.21}$$

When two chain maps ϕ_k and ψ_k induce the *same* homomorphism ϕ_*, the maps ϕ_k and ψ_k are said to be **chain homotopic**. Then there exists a family of homomorphisms $D_k : C_k \to C'_{k+1}$ called **chain homotopy** that satisfy

$$\partial'_{k+1} \circ D_k + D_{k-1} \circ \partial_k = \phi_k - \psi_k \tag{2.22}$$

for each k [48]. Then for any $z \in Z_k$ the k-cycles $\phi_k(z)$ and $\psi_k(z)$ in Z'_k are homologous, since their difference is the boundary $\partial'_{k+1}(D_k(z))$ in B'_k.

Importantly, a chain map $\phi_k : C_k \to C'_k$ is a **chain equivalence** if there exists such chain map $\phi'_k : C'_k \to C_k$ that $\phi'_k \circ \phi_k$ and $\phi_k \circ \phi'_k$ are chain homotopic to the identity maps of C_k and C'_k, respectively. Then the induced maps $\phi_* : H_k(\mathcal{C}) \to H_k(\mathcal{C}')$ and $\phi'_* : H_k(\mathcal{C}') \to H_k(\mathcal{C})$ are isomorphisms and its inverse between the homology groups of the chain complexes $\mathcal{C}$ and $\mathcal{C}'$, i.e. $\phi_*^{-1} = \phi'_*$ holds [48]. ***This observation leads to the possibility of homology-preserving "reduction" of chain complexes by chain equivalences.***

Relative chain complex

Let $\mathcal{C} = (C_\bullet, \partial_\bullet)$ and $\acute{\mathcal{C}} = (\acute{C}_\bullet, \acute{\partial}_\bullet)$ be two chain complexes so that $\acute{C}_k$ is a subgroup of C_k for each k. Let $i_k : \acute{C}_k \to C_k$ be the corresponding inclusion map for each k. The quotient group

$$\bar{C}_k := C_k / \acute{C}_k = \{c + \acute{C}_k \,|\, c \in C_k\} \tag{2.23}$$

is called the **relative k-chain group of C_k modulo $\acute{C}_k$**. That is, the elements $[c]$ of $\bar{C}_k$ are equivalence classes of C_k with the equivalence relation

$$c_1 \simeq c_2 \iff c_1 = c_2 + i(\acute{c}) \tag{2.24}$$

where $c_1, c_2 \in C_k$ and $\acute{c} \in \acute{C}_k$. With the surjection $j : C_k \to \bar{C}_k$, $j(c) = [c]$, the sequence

$$0 \to \acute{C}_k \xrightarrow{i} C_k \xrightarrow{j} \bar{C}_k \to 0 \tag{2.25}$$

is exact. The relative chain groups $\bar{C}_k$ constitute a relative chain complex $\bar{\mathcal{C}} = (\bar{C}_\bullet, \bar{\partial}_\bullet)$, where the boundary operators $\bar{\partial}_k$ are defined to be such that $\bar{\partial}_k[c] := [\partial_k c]$ holds for all $[c] \in \bar{C}_k$ and for each k. The resulting homology groups $H_k(\bar{\mathcal{C}})$ of the relative chain complex are called **relative homology groups**.

Remark 2.1.1. It is possible to choose a $j : C_k \to \bar{C}_k$ that the group $\bar{C}_k$ would have a torsion subgroup. In the applications of this thesis, we will choose such j that the group $\bar{C}_k$ will be torsion-free.

Functoriality

The following pattern will emerge throughout this thesis. Let X and Y be *topological spaces* and let f be a *continuous* map between them. One can associate chain complexes $\mathcal{C}(X) = (C_\bullet(X), \partial_\bullet)$ and $\mathcal{C}(Y) = (C_\bullet(Y), \partial_\bullet)$ with them as described in a later section. Then, f induces a family of chain maps $\phi_k(f)$ between the chain complexes $(C_\bullet(X), \partial_\bullet)$ and $(C_\bullet(Y), \partial_\bullet)$ and thus also homomorphisms $(\phi_k(f))_*$ between the homology groups $H_k(X) := H_k(\mathcal{C}(X))$ and $H_k(Y) := H_k(\mathcal{C}(Y))$.

In *category theory*, such construct that maps both sets and functions between them to other sets and functions in a structure-preserving manner is called a **functor**. In this instance, to each topological space X homology groups $H_k(X)$ are being associated, and each continuous mapping f between topological spaces X and Y induces a homomorphism $(\phi_k(f))_*$ between their homology groups $H_k(X)$ and $H_k(Y)$.

Especially, if f is an injection from a subset $X \subset Y$ to Y, it induces a relative chain complex $\mathcal{C}(Y, X) = (C_\bullet(Y, X), \partial_\bullet)$. Also, if $f : X \to Y$ is a homotopy equivalence [31], the induced homomorphism $(\phi_k(f))_*$ is an isomorphism between the homology groups $H_k(X)$ and $H_k(Y)$. Therefore as we shall see, ***the purely algebraic concept of homology group will reflect topological similarities of the spaces X and Y.***

Cochain complex and cohomology

Let $\mathcal{C}$ be a chain complex and G be an abelian group. The set $C^k(\mathcal{C}; G) := Hom(C_k, G)$ of all homomorphisms $\gamma : C_k \to G$ called **k-cochains** is itself an abelian group when one defines:

$$(\gamma_1 + \gamma_2)(c) = \gamma_1(c) + \gamma_2(c) \quad \forall\, c \in C_k \tag{2.26}$$

$$-\gamma(c) = \gamma(-c) \in G \quad \forall\, c \in C_k \tag{2.27}$$

The boundary homomorphisms $\partial_k : C_k \to C_{k-1}$ induce the **coboundary homomorphisms** $\delta_k : C^k(\mathcal{C}; G) \to C^{k+1}(\mathcal{C}; G)$ defined to operate on k-cochains by:

$$\delta_k \gamma := \gamma \circ \partial_{k+1}. \tag{2.28}$$

That is, $(\delta_k\gamma)(c) = \gamma(\partial_{k+1}c)$ holds for all $k+1$-chains c. The **cochain complex** is the sequence

$$\ldots \xleftarrow{\delta_{k+1}} C^{k+1}(\mathcal{C}, G) \xleftarrow{\delta_k} C^k(\mathcal{C}, G) \xleftarrow{\delta_{k-1}} C^{k-1}(\mathcal{C}, G) \xleftarrow{\delta_{k-2}} \ldots \tag{2.29}$$

of abelian groups linked by the coboundary homomorphisms. The degree k is dropped from the notation δ_k when appropriate.

The elements of the subgroups $Z^k(\mathcal{C}; G) = \ker \delta_k$ and $B^k(\mathcal{C}; G) = \operatorname{im} \delta_{k-1}$ of $C^k(\mathcal{C}; G)$ are called k**-cocycles** and k**-coboundaries**, respectively. The quotient group

$$H^k(\mathcal{C}; G) := Z^k(\mathcal{C}; G)/B^k(\mathcal{C}; G) = \{\zeta + B^k(\mathcal{C}; G) \,|\, \zeta \in Z^k(\mathcal{C}; G)\} \tag{2.30}$$

is called the k**:th cohomology group** of the cochain complex. The **relative cochain complex** and the **relative cohomology groups** are constructed similarly to a relative chain complex.

The elements $[\zeta]$ of the group $H^k(\mathcal{C}; G)$ can be regarded as homomorphisms from the homology group $H_k(\mathcal{C})$ to G. This is seen from the pairing

$$\begin{aligned}(\zeta + \delta_{k-1}\gamma)(z + \partial_{k+1}c) &= \zeta(z) + \zeta(\partial_{k+1}c) + \delta_{k-1}\gamma(z) + \delta_{k-1}\gamma(\partial_{k+1}c) \\ &= \zeta(z) + \delta_k\zeta(c) + \gamma(\partial_k z) + \gamma(\partial_k\partial_{k+1}c) \\ &= \zeta(z) \quad \forall\, \gamma \in C^k(\mathcal{C}; G), c \in C_k(\mathcal{C}).\end{aligned} \tag{2.31}$$

That is, the result is independent of the representatives chosen for $[\zeta]$ and $[z]$. Therefore, we may defined a pairing $H^k(\mathcal{C}; G) \times H_k(\mathcal{C}) \to G$:

$$[\zeta]([z]) := \zeta(z). \tag{2.32}$$

This suggests that the group $Hom(H_k(\mathcal{C}), G)$ of homomorphisms $H_k(\mathcal{C}) \to G$ and the group $H^k(\mathcal{C}; G)$ are related. Indeed, there exists a surjective homomorphism $h_k : H^k(\mathcal{C}; G) \to Hom(H_k(\mathcal{C}), G)$, which is also injective if the groups $H_{k-1}(\mathcal{C})$ and $H_k(\mathcal{C})$ are torsion-free [48, 31].

Any homomorphism $\phi_k : C_k \to C'_k$ induces a homomorphism $\tilde{\phi}_k : C^k(\mathcal{C}'; G) \to C^k(\mathcal{C}; G)$ between the cochain groups in the reverse direction. It is defined by

$$\tilde{\phi}_k(\gamma) = \gamma \circ \phi_k. \tag{2.33}$$

Again, we drop k from the notation it is not needed. If ϕ is a chain map, $\tilde{\phi}$ commutes with the coboundary homomorphism and is thus a chain map. A chain map $\tilde{\phi}$ induces a homomorphism of the cohomology groups in the reverse direction: $\phi^* : H^k(\mathcal{C}'; G) \to H^k(\mathcal{C}; G)$ defined by

$$\phi^*([\zeta]) := [\tilde{\phi}(\gamma)], \quad \zeta \in Z^k(\mathcal{C}'; G). \tag{2.34}$$

The following result makes it possible to use chain equivalences not only for homology-preserving reduction of of chain complexes, but also for cohomology-preserving reduction. Let $\mathcal{C}$ and $\mathcal{C}'$ be chain complexes of free abelian groups. If $H_k(\mathcal{C})$ is isomorphic to $H_k(\mathcal{C}')$, then $H^k(\mathcal{C}; G)$ is isomorphic to $H^k(\mathcal{C}'; G)$ [48]. Consequently, if $\phi : C_k \to C'_k$ is a chain equivalence, then the induced maps $\phi_* : H_k(\mathcal{C}) \to H_k(\mathcal{C}')$ and $\phi^* : H^k(\mathcal{C}'; G) \to H^k(\mathcal{C}; G)$ are isomorphisms.

Summary

The important results from the homological algebra presented in this section are summarized in the following diagram:

$$\begin{array}{ccccccc} H_\bullet(\mathcal{C}) & \longleftarrow & \mathcal{C} & \longrightarrow & C^\bullet(\mathcal{C};G) & \longrightarrow & H^\bullet(\mathcal{C};G) \\ \uparrow\downarrow & & \downarrow & & \uparrow & & \uparrow\downarrow \\ H_\bullet(\mathcal{C}') & \longleftarrow & \mathcal{C}' & \longrightarrow & C^\bullet(\mathcal{C}';G) & \longrightarrow & H^\bullet(\mathcal{C}';G) \end{array} \qquad \begin{array}{c} \phi_* \\ \\ \phi \\ \\ \tilde{\phi} \\ \\ \phi^* \end{array} \tag{2.35}$$

That is, a chain complex $\mathcal{C}$ of free abelian groups C_k induces homology groups $H_\bullet(\mathcal{C})$ and cohomology groups $H^\bullet(\mathcal{C};G)$. If the chain map ϕ is a chain equivalence, then the homomorphisms ϕ_* and ϕ^* are isomorphisms.

2.1.3 Vector space

Linear algebra is perhaps the most applied field of mathematics in computational sciences as "linearization" seems to be an efficient yet accurate enough approximation to model many observed real world phenomena. Linear algebra studies properties of *linear maps* between *vector spaces*. Like finitely generated abelian groups, finite dimensional vector spaces can be endowed with a basis. A basis representation of the elements of a vector space and linear maps turn the problems of abstract algebra into problems of arithmetic, for which a wealth of computational algorithms has been devised.

Let $(V,+)$ be an abelian group and let $\mathbb{F}$ be a *field* of real or complex numbers. They form a **vector space** when one defines a map $\cdot : \mathbb{F} \times V \to V$ with the familiar requirements, see for example [25]. As customary, the map $\cdot$ is dropped from the notation and the vector space is denoted with plain V.

Let $S = \{s_1, s_2, \ldots, s_n\}$ be a finite subset of V. If any element v of V can be written as a *linear combination* $v = \sum_{i=1}^n v^i s_i$, S is said to **span** V. If $\sum_{i=1}^n v^i s_i = 0$ implies $v^1 = v^2 = \ldots = v^n = 0$, the set S is said to be **linearly independent**. A linearly independent subset $S \subset V$ that spans V is called a **basis** of V, and $n = |S|$ is the **dimension** of V. Every finite dimensional vector space has a basis [25]. To define a basis for an infinite dimensional vector space some additional structure is required, since an infinite sum bears no meaning without any additional structure, for example *topology* or *norm* on V.

An element v of n-dimensional vector space V has a basis representation $\mathbf{v} \in \mathbb{F}^n$, and we denote $v = \sum_{i=1}^n v^i s_i = (s_1, s_2, \ldots, s_n)\mathbf{v} = \mathrm{S}\mathbf{v}$.

If a map $\phi : V \to W$ is a vector space homomorphism, it is called a **linear map**. That is, for all $c_1, c_2 \in \mathbb{F}$ and $v_1, v_2 \in V$, $\phi(c_1 v_1 + c_2 v_2) = c_1\phi(v_1) + c_2\phi(v_2)$ holds. Let $\mathrm{S} = (s_1, s_2, \ldots, s_n)$ and $\mathrm{R} = (r_1, r_2, \ldots, r_m)$ be bases for V and W, respectively. Then, $\phi(s_i) = \mathrm{R}\boldsymbol{\phi}_i$ for some $\boldsymbol{\phi}_i \in \mathbb{F}^m$. Consequently, an $m \times n$ matrix $\mathrm{F} = \left[\boldsymbol{\phi}_1, \boldsymbol{\phi}_2, \ldots, \boldsymbol{\phi}_n\right]$ represents a linear map ϕ. If $\dim V = \dim W$ holds and ϕ is an isomorphism, the matrix F is invertible.

The **dual space** V^* of the vector space V is a vector space of linear maps from V to $\mathbb{F}$. The vector space structure for V^* is obtained with the definitions

$$(\omega + \eta)(v) := \omega(v) + \eta(v), \quad \omega, \eta \in V^*, v \in V \tag{2.36}$$

$$(c\omega)(v) := c\omega(v). \tag{2.37}$$

Since a basis for the vector space $\mathbb{F}$ is just a single non-zero element $r \in \mathbb{F}$, the elements $\omega \in V^*$ can be represented by vectors $\boldsymbol{\omega} \in \mathbb{F}^n$. Then, the evaluation is given by $\omega(v) = \boldsymbol{\omega}^T \mathbf{v} \in \mathbb{F}$.

A linear map $\phi : V \to W$ induces a map $\phi^* : W^* \to V^*$ defined by

$$\phi^*\omega(v) := (\omega \circ \phi)(v) = \omega(\phi(v)) \quad \forall\, \omega \in W^*, v \in V. \tag{2.38}$$

The map ϕ^* is called the **pullback** of the map ϕ. If ϕ is represented by the matrix F in some bases of V and W, one can read from the definition of the pullback ϕ^* that it is represented by the transpose matrix F^T. From the above discussion one can infer a structure represented by the commutative diagrams:

$$\begin{array}{ccc} V & \xrightarrow{\phi} & W \\ \updownarrow & & \updownarrow \\ \mathbb{F}^n & \xrightarrow{\mathrm{F}} & \mathbb{F}^m \end{array} \quad \text{and} \quad \begin{array}{ccc} V^* & \xleftarrow{\phi^*} & W^* \\ \updownarrow & & \updownarrow \\ \mathbb{F}^n & \xleftarrow{\mathrm{F}^T} & \mathbb{F}^m \end{array} \tag{2.39}$$

Let $\langle \cdot, \cdot \rangle : V \times V \to \mathbb{F}$ denote an *inner product* on n-dimensional vector space V. Given a basis for V, the inner product is represented by a symmetric or hermitian $n \times n$ matrix G. The inner product is then evaluated by $\langle v_1, v_2 \rangle = \mathbf{v}_1^H \mathrm{G} \mathbf{v}_2$, where $\mathbf{v}_1$, $\mathbf{v}_2$ represent v_1, $v_2 \in V$ in the basis and $\mathbf{v}_1^H$ denotes conjugate transpose of $\mathbf{v}_1$. A basis $(s_1, s_2, \ldots, s_n)$ of V is **orthonormal**, if $\langle s_i, s_j \rangle = \delta_{ij}$ holds for all i, j.

Let V be a *complete* inner product space, i.e. a *Hilbert space* [25]. Then, if $\omega : V \to \mathbb{F}$ is a linear map, i.e. $\omega \in V^*$, then there exists an unique vector $w \in V$ such that $\omega(v) = \langle v, w \rangle$ holds for all $v \in V$ [25, 23]. That is, ***the inner product on a Hilbert space*** V ***induces an isomorphism***[1] ***from*** V ***to*** V^* by Riesz representation theorem. Given a basis for V and V^*, and the corresponding inner product matrix G, the isomorphism is given by $\boldsymbol{\omega} = \mathrm{G}\mathbf{v}$, where $\boldsymbol{\omega}$ and $\mathbf{v}$ represent ω and v, respectively. The isomorphism $V \to V^*$ defined by an inner product on V and its inverse are sometimes called **flat** $v^\flat \in V^*$ and **sharp** $\omega^\sharp \in V$, respectively. The isomorphism also induces an inner product on the dual space V^* given by the sharp:

$$\langle \omega, \eta \rangle := \langle \omega^\sharp, \eta^\sharp \rangle = (\mathrm{G}^{-1}\boldsymbol{\omega})^H \mathrm{G} \mathrm{G}^{-1} \boldsymbol{\eta} = \boldsymbol{\omega}^T \mathrm{G}^{-1} \boldsymbol{\eta}. \tag{2.40}$$

Let $\phi : V \to W$ be a linear map. Given a basis for W the map ϕ has a corresponding matrix representation F. If one requires that $\langle v_1, v_2 \rangle = \langle \phi(v_1), \phi(v_2) \rangle$ must hold for all $v_1, v_2 \in V$, one infers that $\mathbf{v}_1^H \mathrm{G} \mathbf{v}_2 = (\mathrm{F}\mathbf{v}_1)^H \mathrm{H} \mathrm{F} \mathbf{v}_2 = \mathbf{v}_1^H \mathrm{F}^T \mathrm{H} \mathrm{F} \mathbf{v}_2$ hold. That is, if the inner product $\langle \cdot, \cdot \rangle_W$ on W is represented by the matrix H, its corresponding "pulled back" inner product on V is represented by the matrix $\mathrm{F}^T\mathrm{HF}$. In the special case when $\dim V = \dim W$ holds and ϕ is an isomorphism, this is the change of basis formula for the inner product matrices.

[1] There are also other ways to induce an isomorphism between a vector space and its dual.

Homology and cohomology vector space

Homology and cohomology groups can be interpreted as vector spaces, but some information about them may be lost in the process.

Let $H_k(\mathcal{C})$ and $H^k(\mathcal{C}; G)$ be finitely generated and let the abelian groups C_k in $\mathcal{C}$ be free. Denote by $H_k(\mathcal{C}; \mathbb{F})$ a homology *vector space* which is obtained from the homology group by allowing the group generators have infinite field of characteristic zero coefficients instead of integer coefficients[2]. That is,

$$H_k(\mathcal{C}; \mathbb{F}) := \text{span}\,\{[z_i]\}, \quad [z_i] \in S, \tag{2.41}$$

where S is a generating set of the group $H_k(\mathcal{C})$. The dimension of the homology space is equal to the rank of the free subgroup F of the homology group. That is, if $[z_i]$ is a torsion element of $H_k(\mathcal{C})$, it can be removed from the spanning set of $H_k(\mathcal{C}; \mathbb{F})$ since it is linearly dependent on the others. Therefore, ***when one interprets a homology group as a vector space, the information about its torsion subgroup is lost.*** The dimension of the k:th homology space is called the k:th betti number $\beta_k(\mathcal{C})$:

$$\dim H_k(\mathcal{C}; \mathbb{F}) := \beta_k(\mathcal{C}) = \text{rank}\, F \leq |S|. \tag{2.42}$$

In particular, if $H_k(\mathcal{C})$ is torsion-free, $\beta_k(\mathcal{C}) = |S|$ holds.

The cohomology group $H^k(\mathcal{C}; \mathbb{F})$ is readily a *vector space*. However, it is a nontrivial fact that the vector space $Hom(H_k(\mathcal{C}; \mathbb{F}), \mathbb{F})$ of all linear maps $H_k(\mathcal{C}; \mathbb{F}) \to \mathbb{F}$ is isomorphic to the vector space $H^k(\mathcal{C}; \mathbb{F})$ [48]. In contrast, the *group* of homomorphisms $H_k(\mathcal{C}) \to G$ is *not* isomorphic to the group $H^k(\mathcal{C}; G)$ in general. With the isomorphism $Hom(H_k(\mathcal{C}; \mathbb{F}), \mathbb{F}) \simeq H^k(\mathcal{C}; G)$, the cohomology space $H^k(\mathcal{C}; \mathbb{F})$ turns out to be the dual space of the homology space $H_k(\mathcal{C}; \mathbb{F})$.

2.1.4 Exterior algebra

Exterior algebra has its roots in the geometric interpretation of vectors. For example, two vectors that are linearly independent span a plane. If the vectors are linearly dependent, the spanned plane reduces to a line and thus has zero surface area. If one interchanges the order of the spanning vectors, the plane may be thought to have the opposite orientation. A volume generated by three vectors behaves similarly. This concept to generate geometric entities from vectors is captured by the properties of the *exterior product*.

There's also a concept that is dual to this. One can think that one assigns quantities to the geometric entities spanned by vectors. So-called *alternating multilinear maps* from an ordered set of vectors to scalar values have the desired properties. If the same vector is more than once as an input argument, the result is zero. If the order of two input vectors is interchanged, the result changes its sign as if the orientation was changed.

Let V be a real or complex vector space over $\mathbb{F}$ of dimension n. A k-linear map $\omega : V \times \ldots \times V \to \mathbb{F}$ is **alternating** if

$$\omega(\ldots, v_i, \ldots v_j, \ldots) = -\omega(\ldots, v_j, \ldots v_i, \ldots) \tag{2.43}$$

[2]The most natural framework for this result is *modules*. Abelian groups are modules over the ring of integers, and vector spaces are modules over a field. The *Universal Coefficient Theorem* [31] is the generalization of the following result in the framework of modules.

holds for each pair of entries $v_i, v_j \in V$. The space of alternating k-linear maps is denoted by $\Lambda^k(V^*)$. The space $\Lambda^1(V^*)$ coincides with the dual space V^* of V and $\Lambda^0(V^*)$ coincides with $\mathbb{F}$. Similarly, the space of alternating k-linear maps $V^* \times \ldots \times V^* \to \mathbb{F}$ is denoted by $\Lambda^k(V)$ and the space $\Lambda^1(V)$ coincides with V. The elements of the spaces $\Lambda^k(V)$ and $\Lambda^k(V^*)$ are called **k-vectors** and **k-covectors**, respectively.

Because of the alternating property, a covector $\omega \in \Lambda^k(V^*)$ is completely determined by its values with a basis $(s_1, s_2, \ldots, s_n)$ of V so that in $\omega(s_{i_1}, s_{i_2} \ldots, s_{i_k})$ the indices i are always in strictly increasing order. Therefore, $\dim \Lambda^k(V^*) = \binom{n}{k}$ holds [24].

We adopt the **multi-index** notation to simplify the expressions. Multi-index I is an ordered m-tuple $I = (i_1, i_2, \ldots, i_m)$. Denote by $\vec{I}$ such multi-index I for which $i_1 < i_2 < \ldots < i_m$ holds. Also define a **Kronecker delta for multi-indices** I and J to be such that

$$\delta_I^J = \begin{cases} 1 & \text{if } I = (i_1, i_2, \ldots, i_m) \text{ is an } \textit{even} \text{ permutation of } J = (j_1, j_2, \ldots, j_k). \\ -1 & \text{if } I \text{ is an } \textit{odd} \text{ permutation of } J. \\ 0 & \text{if } I \text{ is not a permutation of } J. \end{cases} \tag{2.44}$$

The **exterior product** $\wedge$ of covectors is a map $\wedge : \Lambda^k(V^*) \times \Lambda^l(V^*) \to \Lambda^{k+l}(V^*)$ is defined by

$$(\omega \wedge \eta)(v_{\vec{I}}) := \sum_{\vec{K}} \sum_{\vec{J}} \delta_I^{JK} \omega(v_{\vec{J}}) \eta(v_{\vec{K}}), \tag{2.45}$$

where JK denotes a multi-index that is the concatenation of the indices of J and K and $v_{\vec{I}}$ denotes a tuple of elements of V indexed by the multi-index $\vec{I}$. The exterior product of vectors $\wedge : \Lambda^k(V) \times \Lambda^l(V) \to \Lambda^{k+l}(V)$ is defined similarly.

Given a basis $(s_1, s_2, \ldots, s_n)$ for V, denote a representation of its element $v = \sum_{i=1}^n v^i s_i$ by $\mathbf{v} = [v^1, v^2, \ldots, v^n]^T \in \mathbb{F}^n$. The space $\Lambda^k(V)$ has a basis $(s_{\vec{I}})_{\vec{I}} = (s_{i_1} \wedge s_{i_2} \wedge \ldots \wedge s_{i_k})_{\vec{I}}$. An element ω of $\Lambda^k(V^*)$ can be represented by $\boldsymbol{\omega} \in \mathbb{F}^{\binom{n}{k}}$ so that $\omega = \sum_{\vec{I}} \omega_{\vec{I}} \sigma^{\vec{I}}$ holds, where $\sigma^{\vec{I}} = \sigma^{i_1} \wedge \sigma^{i^2} \wedge \ldots \wedge \sigma^{i_k}$ is such that $\sigma^{\vec{I}}(s_{j_1}, s_{j_2}, \ldots, s_{j_k}) = \sigma^{\vec{I}}(s_{\vec{J}}) = \delta_J^I$ holds. The exterior product induces a bilinear pairing $\Lambda^k(V^*) \times \Lambda^l(V) \to \mathbb{F}$ such that $\omega(v) = \boldsymbol{\omega}^T \mathbf{v}$ holds.

Let $\phi : V \to W$ be a linear map between vector spaces with $\dim V = n$ and $\dim W = m$. The exterior product induces a linear map $\Lambda^k \phi^* : \Lambda^k(W^*) \to \Lambda^k(V^*)$ of k-covectors defined by

$$(\Lambda^k \phi^*)(\tau^{i_1} \wedge \tau^{i_2} \wedge \ldots \wedge \tau^{i_k}) := \phi^* \tau^{i_1} \wedge \phi^* \tau^{i^2} \wedge \ldots \wedge \phi^* \tau^{i_k} \tag{2.46}$$

for the basis elements $\tau^{\vec{I}}$ of $\Lambda^k(W^*)$. The definition extends to all $\omega \in \Lambda^k(W^*)$ by the representation $\omega = \sum_{\vec{I}} \omega_{\vec{I}} \tau^{\vec{I}}$. The map $\Lambda^k \phi^*$ is also called the pullback and abbreviated to ϕ^* as the value of k is often clear from the context. Let $\sigma^{\vec{J}}$ form a basis for $\Lambda^k(V^*)$. If the map ϕ is represented by an $m \times n$ matrix F, the pullback $\Lambda^k \phi^*$ is represented by a matrix whose elements are determinants of the $k \times k$ matrices $(\mathrm{F}^T)_{\vec{J}}^{\vec{I}}$ which contain the rows $\vec{I}$ and the columns $\vec{J}$ of the matrix F^T [24, 23]. That is,

$$\phi^* \omega = \sum_{\vec{I}} \omega_{\vec{I}} \phi^* \tau^{\vec{I}} = \sum_{\vec{J}} \sum_{\vec{I}} \omega_{\vec{I}} \det((\mathrm{F}^T)_{\vec{J}}^{\vec{I}}) \sigma^{\vec{J}} = \sum_{\vec{J}} \omega_{\vec{J}} \sigma^{\vec{J}} \in \Lambda^k(V^*), \tag{2.47}$$

where $\omega_{\vec{J}} = \sum_{\vec{I}} \omega_{\vec{I}} \det((\mathrm{F}^T)^{\vec{I}}_{\vec{J}})$ represent ω in the basis of $\Lambda^k(V^*)$. Similarly, the map ϕ induces a linear map $\Lambda^k\phi : \Lambda^k(V) \to \Lambda^k(W)$ of k-vectors given by

$$\phi(v) = \sum_{\vec{I}} v^{\vec{I}} \phi(s_{\vec{I}}) = \sum_{\vec{J}} \sum_{\vec{I}} v^{\vec{I}} \det(\mathrm{F}^{\vec{J}}_{\vec{I}}) r_{\vec{J}} = \sum_{\vec{J}} v^{\vec{J}} r_{\vec{J}} \in \Lambda^k(W), \tag{2.48}$$

where $s_{\vec{I}}$ and $r_{\vec{J}}$ form bases for $\Lambda^k(V)$ and $\Lambda^k(V)$, respectively [24, 23].

The inner product on V induces inner products for the spaces $\Lambda^k(V^*)$. Let $\sigma^{\vec{I}}$ form a basis for $\Lambda^k(V^*)$ and define

$$\langle \sigma^{\vec{I}}, \sigma^{\vec{J}} \rangle := \begin{vmatrix} \langle \sigma^{i_1}, \sigma^{j_1} \rangle & \langle \sigma^{i_1}, \sigma^{j_2} \rangle & \dots & \langle \sigma^{i_1}, \sigma^{j_k} \rangle \\ \langle \sigma^{i_2}, \sigma^{j_1} \rangle & \langle \sigma^{i_2}, \sigma^{j_2} \rangle & \dots & \langle \sigma^{i_2}, \sigma^{j_k} \rangle \\ \vdots & \vdots & \ddots & \vdots \\ \langle \sigma^{i_k}, \sigma^{j_1} \rangle & \langle \sigma^{i_k}, \sigma^{j_2} \rangle & \dots & \langle \sigma^{i_k}, \sigma^{j_k} \rangle \end{vmatrix} = \det((\mathrm{G}^{-1})^{\vec{I}\vec{J}}), \tag{2.49}$$

where the matrix $(\mathrm{G}^{-1})^{\vec{I}\vec{J}}$ contains the rows $\vec{I}$ and columns $\vec{J}$ of the inverse of the inner product matrix G of V. Then, the inner product of $\omega = \sum_{\vec{I}} \omega_{\vec{I}} \sigma^{\vec{I}}$ and $\eta = \sum_{\vec{J}} \eta_{\vec{J}} \sigma^{\vec{J}}$ is given by

$$\langle \omega, \eta \rangle = \sum_{\vec{I}} \sum_{\vec{J}} \omega_{\vec{I}} \eta_{\vec{J}} \langle \sigma^{\vec{I}}, \sigma^{\vec{J}} \rangle. \tag{2.50}$$

Again, an inner product for the space $\Lambda^k(V)$ is induced similarly. One defines

$$\langle v, w \rangle = \sum_{\vec{I}} \sum_{\vec{J}} v^{\vec{I}} w^{\vec{J}} \langle s_{\vec{I}}, s_{\vec{J}} \rangle = \sum_{\vec{I}} \sum_{\vec{J}} v^{\vec{I}} w^{\vec{J}} \det(\mathrm{G}_{\vec{I}\vec{J}}). \tag{2.51}$$

Let $(\sigma^1, \sigma^2, \dots, \sigma^n)$ be an orthonormal basis for V. A linear map called **Hodge operator** $\star : \Lambda^k(V^*) \to \Lambda^{n-k}(V^*)$ is defined to be the one that satisfies

$$\eta \wedge \star\omega = \langle \eta, \omega \rangle \sigma^1 \wedge \sigma^2 \wedge \ldots \wedge \sigma^n \tag{2.52}$$

for all $\eta, \omega \in \Lambda^k(V^*)$. If $(\tau^1, \tau^2, \dots, \tau^n)$ is a non-orthonormal basis and ϕ is a change-of-basis map, one has $\sigma^1 \wedge \sigma^2 \wedge \ldots \wedge \sigma^n = \phi^*(\tau^1 \wedge \tau^2 \wedge \ldots \wedge \tau^n) = \det(\mathrm{F}^T)\tau^1 \wedge \tau^2 \wedge \ldots \wedge \tau^n$.

A similar map $\star : \Lambda^k(V) \to \Lambda^{n-k}(V)$ for k-vectors is defined analogously. The Hodge operator satisfies $\star\star\omega = (-1)^{k(n-k)}\omega$, i.e. expect for the sign, it is its own inverse.

2.2 Manifold and its cell decomposition

In a modern treatise, a boundary value problem is established on a *manifold*. It is a topological space that can be addressed with real number coordinates. However, the actual *choice* of coordinates is left open. That is, most of the analysis is done without any reference to some specific choice of coordinates. It is enough to know that at a will, a coordinate system is available.

We first define the real coordinate space $\mathbb{R}^n$ and the Euclidean space $\mathbb{E}^n$, and discuss how they are related. The precise concept of manifold can be build upon these ingredients.

2.2.1 Real coordinate space

The **real coordinate space** $\mathbb{R}^n$ is an n-dimensional vector space of n-tuples of real numbers over $\mathbb{R}$. The **half-space** $\mathbb{R}^n_+$ is the *set*

$$\mathbb{R}^n_+ := \{x = (x_1, x_2, \ldots, x_n) \in \mathbb{R}^n \,|\, x_n \geq 0\}. \tag{2.53}$$

Its boundary is the set $\partial\mathbb{R}^n_+ = \{x = (x_1, x_2, \ldots, x_n) \in \mathbb{R}^n \,|\, x_n = 0\}$.

Let the *ordered* sets $(\mathbf{e}_1, \mathbf{e}_2, \ldots, \mathbf{e}_n)$ and $(\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_n)$ be the standard basis and a basis of $\mathbb{R}^n$, respectively. Then there exists a linear map $\mathrm{A} : \mathbb{R}^n \to \mathbb{R}^n$ that satisfies

$$\mathrm{A}\mathbf{e}_i = \mathbf{v}_i \quad \forall\, i \in \{1, 2, \ldots, n\} \tag{2.54}$$

If the determinant of the matrix A is positive, the basis $(\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_n)$ has the **positive orientation**, otherwise it has a **negative orientation**. The orientation is denoted by $o = \pm 1$. The oriented real coordinate space is a pair $(\mathbb{R}^n, o)$.

2.2.2 Euclidean space

The Euclidean space is a model for affine and metric properties of our environment, such as translation of rigid objects, and measurement of distances and angles. Structurally, it is an *affine space* together with metric properties that are induced by Cartesian coordinate system.

The **affine space** is a pair (A, V) of a set A and a vector space V together with a map $t : V \times A \to A$ denoted by $t(v, a) = v + a$. It is required that $0 + a = 0$ and $(v + w) + a = v + (w + a)$ hold for all $a \in A$ and $v, w \in V$. Also, with $a \in A$ fixed, the map $V \to A : t_a(v) = v + a$ is required to be a bijection.

The **Euclidean space** $\mathbb{E}^n$ is an affine space (A, V) where the n-dimensional vector space V has an inner product $\langle \cdot, \cdot \rangle$ and A has a **metric** defined by $d(a, a + v) := \sqrt{\langle v, v \rangle}$. The usual way to construct the Euclidean space $\mathbb{E}^n$ is to set $A = V = \mathbb{R}^n$ and define

$$t(\mathbf{v}, \mathbf{a}) := \mathbf{v} + \mathbf{a} \quad \text{and} \quad \langle \mathbf{v}, \mathbf{v} \rangle := \mathbf{v}^T\mathbf{v}. \tag{2.55}$$

That is, the real coordinate space $\mathbb{R}^n$ is assumed to be a Cartesian coordinate system for the Euclidean space $\mathbb{E}^n$.

The inner product can be used to induce both norm $\|\mathbf{v}\| := \sqrt{\langle \mathbf{v}, \mathbf{v} \rangle}$ and topology τ for the spaces $\mathbb{E}^n$ and $\mathbb{R}^n$. A *basis for the topology* is formed by the Euclidean n-balls: An open y-centered r-radius **Euclidean n-ball** $B^n_r(y)$ is a set

$$B^n_r(y) := \{x \in \mathbb{R}^n \,|\, \|x - y\| < r\}. \tag{2.56}$$

The set $\mathbb{R}^n_+ \subset \mathbb{R}^n$ can be equipped with the *subspace topology* of the topology defined by $\tau_+ := \{U \cap \mathbb{R}^n_+ \,|\, U \subset \tau\}$. A basis for such topology is formed by "sliced" Euclidean n-balls $B^n_r(y) \cap \mathbb{R}^n_+$.

Remark 2.2.1. The topological spaces $(\mathbb{R}^n, \tau)$ and $(\mathbb{R}^n_+, \tau_+)$ are *Hausdorff spaces*.

Remark 2.2.2. The topological vector space $(\mathbb{R}^n, \tau)$ is an object on its own right, without an inner product, norm, or affine structure of the Euclidean space $\mathbb{E}^n$.

2.2.3 Manifolds

A differentiable n-dimensional manifold M is a topological space that is everywhere *locally homeomorphic* to the topological space $(\mathbb{R}^n, \tau)$ together with *differentiable atlas* of coordinate *charts*.

Charts and atlases

Consider a topological space M with a covering $M = \bigcup_i U_i$ of open sets U_i together with homeomorphisms $x_i : U_i \to V \subset (\mathbb{R}^n, \tau)$. A pair (U_i, x_i) is called a **chart**. See figure 2.1, left. If $M = U$ holds for some chart (U, x) of M, it is called a **global chart**, or global parametrization, and denoted (M, x).

If the **transition maps** $x_i \circ x_j^{-1} : x_j(U_i \cap U_j) \to x_i(U_i \cap U_j)$ are differentiable for all i and j the collection $\mathcal{D} = \{(U_i, x_i)\}$ of charts is called a **differentiable atlas** for M. A pair $(M, \mathcal{D})$ is called a **differentiable n-manifold**.

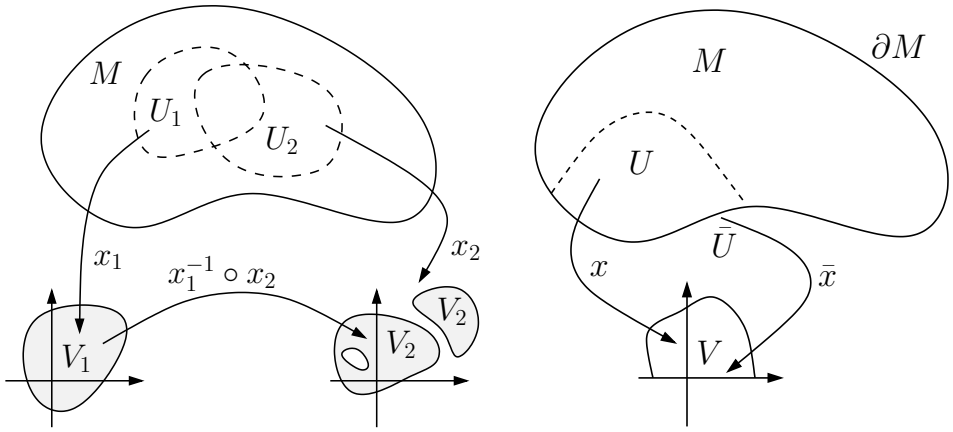


Figure 2.1: Left: A differentiable manifold manifold $(M, \mathcal{D})$ and two of its charts (U_1, x_1) and (U_2, x_2). Right: A differentiable manifold-with-boundary M and a chart $(\bar{U}, \bar{x})$ for the boundary manifold ∂M.

One can describe a differentiable manifold by selecting a suitable atlas for it. However, a differentiable manifold has no privileged atlas, for a different atlas can describe the same differentiable manifold in the following sense: Let the collections $\mathcal{D}_1 = \{(U_i, x_i)\}$ and $\mathcal{D}_2 = \{(V_i, y_i)\}$ qualify as two atlases for a topological space M. The **atlases $\mathcal{D}_1$ and $\mathcal{D}_2$ are equivalent** if the set union $\mathcal{D}_1 \cup \mathcal{D}_2$ is a differentiable atlas. If the atlases are equivalent, they describe the same differentiable manifold. The union of all equivalent atlases for the manifold is called the **maximal atlas**. Later, we denote a differentiable manifold with plain M. A differentiable manifold is called **C^k-manifold**, if all the maps $x_i \circ y_i^{-1}$ in the maximal atlas are k-times continuously differentiable. **Smooth manifold** is a C^k-manifold[3]. for all $k \in \mathbb{N}$.

[3]One also obtains a **C^0-manifold**, i.e. topological manifold when transitions maps are homeomorphisms, and **PL-manifold** or **$PDIFF$-manifold** when the transition maps are *piecewise linear functions* or *piecewise differentiable functions*, respectively. As a generalization in the category theory, one obtains a different *category of manifolds* by changing the class of the transition maps.

Manifold with boundary

A **differentiable n-manifold-with-boundary** agrees with the definition of a differentiable n-manifold, except that the chart maps are required to map to the open sets of $\mathbb{R}^n_+$ in its subspace topology. That is, for $p \in U_i \subset M$, $x_i(p) \in V \subset \mathbb{R}^n_+$ holds, where $V \in \tau_+$. Since one can always choose an equivalent atlas for a manifold such that the chart maps x_i map to the half-space $\mathbb{R}^n_+$ instead of $\mathbb{R}^n$, we make no distinction between a manifold and a manifold-with-boundary later in this thesis. Manifolds-without-boundary are just manifolds-with-boundary which have no *boundary points*.

A point $p \in U \subset M$ is a **boundary point** of the manifold M, if it is mapped to $\partial\mathbb{R}^n_+$ by a chart $(U, x) \in \mathcal{D}$. The **boundary ∂M of a manifold** M is the set of all the boundary points:

$$\partial M := \{p \in M \,|\, x(p) \in \partial\mathbb{R}^n_+ \text{ when } p \in U\}. \tag{2.57}$$

The boundary ∂M is a differentiable $n-1$-manifold that has no boundary points [29]. If $p \in U$ is a boundary point of M and (U, x) is a chart of M, the manifold ∂M inherits a chart $(\bar{U}, \bar{x})$ such that $\bar{x} : \bar{U} \to V \subset \mathbb{R}^{n-1}$ where $\bar{U} = U \cap \partial M$ holds and V is open. It is defined by

$$\bar{x}(p) := (x^1(p), x^2(p), \ldots, x^{n-1}(p)) \in \mathbb{R}^{n-1} \text{ when } x(p) = (x^1(p), x^2(p), \ldots, x^n(p)) \in \mathbb{R}^n_+. \tag{2.58}$$

See figure 2.1, right. We need such inherited chart in the following sections in the definition of boundary relations of *cells* on the manifold.

Tangent space and cotangent space

A map $f : M \to \mathbb{F}$ is differentiable at a point $p \in M$ if the function $f \circ x^{-1} : \mathbb{R}^n \to \mathbb{F}$ is differentiable for every chart (U, x) where $p \in U$. That is, all the partial derivatives $\frac{\partial(f \circ x^{-1})}{\partial x^j}$ exist at $x(p) = (x^1, x^2, \ldots, x^n)$. If $\mathbb{F} = \mathbb{C}$ holds, one has $f = \mathrm{Re}(f) + i\mathrm{Im}(f)$, where $\mathrm{Re}(f), \mathrm{Im}(f) : M \to \mathbb{R}$. Then, f is differentiable if $\mathrm{Re}(f)$ and $\mathrm{Im}(f)$ are differentiable. Denote the space of q-times differentiable functions on the manifold M by $C^q(M; \mathbb{F})$. Maps that are q-times differentiable for all $q \in \mathbb{N}$ are called **smooth**. The chart map $x : M \to \mathbb{R}^n$ can be considered as an n-tuple of differentiable **coordinate maps** $x^i : M \to \mathbb{R}$ such that $x(p) = (x^1(p), x^2(p), \ldots, x^n(p)) \in \mathbb{R}^n$ holds.

A **tangent vector** v_p of a differentiable manifold $(M, \mathcal{D})$ is a directional derivative operator of maps $f : M \to \mathbb{F}$ that are differentiable with respect to the atlas $\mathcal{D}$ at a point $p \in M$. Using a chart x it is defined by

$$v_p(f) := \sum_{i=1}^{n} v_x^i \left.\frac{\partial(f \circ x^{-1})}{\partial x^i}\right|_{x(p)}, \quad \forall\, f \in C^q(M), \tag{2.59}$$

where the coefficient vector $\mathbf{v}_x = (v_x^1, v_x^2, \ldots, v_x^n) \in \mathbb{F}^n$ associated to the chart x gives the direction and the magnitude of the directional derivative in terms of the chart (U, x). Thus one may denote

$$v_p := \sum_{i=1}^{n} v_x^i \frac{\partial}{\partial x^i}. \tag{2.60}$$

Let (V, y) be another chart for the manifold M at p. On that chart, a tangent vector v'_p is given by

$$v'_p := \sum_{i=1}^{n} v_y^i \frac{\partial}{\partial y^i}. \tag{2.61}$$

These two tangent vectors are equivalent if $v_p(f) = v'_p(f)$ holds for all $f \in C^1(M; \mathbb{F})$ at p. That is, if the directional derivative of a function at the point $p \in M$ does not depend on the chart it is evaluated on:

$$v_p(f) = \sum_{i=1}^{n} v_x^i \left.\frac{\partial(f \circ x^{-1})}{\partial x^i}\right|_{x(p)} = \sum_{i=1}^{n} v_y^i \left.\frac{\partial(f \circ y^{-1})}{\partial y^i}\right|_{y(p)} = v'_p(f). \tag{2.62}$$

Let $f = y^j$ hold for all $j \in \{1, 2, , \ldots, n\}$ to obtain

$$v_y^j = \sum_{i=1}^{n} v_x^i \left.\frac{\partial(y^j \circ x^{-1})}{\partial x^i}\right|_{x(p)}. \tag{2.63}$$

That is, $\mathbf{v}_y = \mathrm{J}\mathbf{v}_x$ holds, where J is the real Jacobian $n \times n$ matrix of the transition map $y \circ x^{-1}$ at p.

The set of all tangent vectors v_p of the manifold M at a point p is a vector space denoted by T_pM , called the **tangent space** of M at p. It is isomorphic to $\mathbb{F}^n$, as each chart (U, x) at p picks an isomorphism. The set $\{\partial/\partial x^1, \partial/\partial x^2, \ldots, \partial/\partial x^n\}$ is a basis for T_pM called the **coordinate basis**.

If one considers the function f to be fixed in the directional derivative, one obtains a map $\mathrm{d}f : T_pM \to \mathbb{F}$. It is an element of the the dual space T_p^*M of T_pM, called the **cotangent space**. Let us test what happens when $f = x^j$:

$$\mathrm{d}x^j(v_p) := v_p(x^j) = \sum_{i=1}^{n} v_x^i \left.\frac{\partial(x^j \circ x^{-1})}{\partial x^i}\right|_{x(p)} = v_x^j. \tag{2.64}$$

That is, $\mathrm{d}x^j(\frac{\partial}{\partial x^i}) = \delta_i^j$ holds for all i, j. This observation qualifies the set $\{\mathrm{d}x^1, \mathrm{d}x^2, \ldots, \mathrm{d}x^n\}$ as a basis for T_p^*M called the **coordinate cobasis**. Then, $\mathrm{d}f$ has the following representation in the coordinate cobasis:

$$\mathrm{d}f = \sum_{i=1}^{n} \left.\frac{\partial(f \circ x^{-1})}{\partial x^i}\right|_{x(p)} \mathrm{d}x^i := \sum_{i=1}^{n} \omega_i^x \mathrm{d}x^i. \tag{2.65}$$

That is, the cotangent vector $\mathrm{d}f$ is represented by the coefficient vector $\boldsymbol{\omega}_x \in \mathbb{F}^n$. The evaluation $\mathrm{d}f(v_p)$ should yield the same real or complex number regardless of the basis representations of $\mathrm{d}f$ and v_p. From

$$\mathrm{d}f(v_p) = \sum_{i=1}^{n} \omega_i^x \mathrm{d}x^i(v_p) = \sum_{i=1}^{n} \omega_i^x v_x^i = \boldsymbol{\omega}_x^T \mathbf{v}_x = \boldsymbol{\omega}_x^T \mathrm{J}\mathbf{v}_y = (\mathrm{J}^T\boldsymbol{\omega}_x)^T \mathbf{v}_y \tag{2.66}$$

one deduces that such property is enforced by requiring that $\boldsymbol{\omega}_y = \mathrm{J}^T\boldsymbol{\omega}_x$ holds for the representation $\boldsymbol{\omega}_y$ of $\mathrm{d}f$ on the chart (V, y).

Remark 2.2.3. If $\mathbb{F}^n = \mathbb{C}^n$ holds, the evaluation $\mathrm{d}f(v_p)$ is ***not*** the complex inner product of complex vectors, as in the expression $\boldsymbol{\omega}_x^T\mathbf{v}_x$ one has the ordinary transpose and not the conjugate transpose.

Orientable manifold

As the tangent spaces T_pM are vector spaces that are isomorphic to $\mathbb{R}^n$ via a chart (U, x), one may choose a chart at $p \in U$ for which one assigns a positive orientation, and let (V, y) be an another chart with $p \in V$. The two bases $\{\partial/\partial x^1, \partial/\partial x^2, \ldots, \partial/\partial x^n\}$ and $\{\partial/\partial y^1, \partial/\partial y^2, \ldots, \partial/\partial y^n\}$ of T_pM have the same orientation, if the determinant of the Jacobian matrix of the transition map $y \circ x^{-1}$ is positive, i.e. if $\det \mathrm{J} > 0$ holds.

A differentiable n-manifold is **orientable**, if it has an atlas $\mathcal{D} = \{(U_i, x_i)\}$ of charts such that for all $p \in U_i \cap U_j$ with all chart domains U_i, U_j in $\mathcal{D}$, the bases $\{\partial/\partial x_i^k\}_{k=1}^n$ and $\{\partial/\partial y_j^k\}_{k=1}^n$ of T_pM have the same orientation. If such atlas does not exist, the manifold is **nonorientable**. An **oriented differentiable n-manifold** is a pair $(M, \mathcal{D})$, where all the charts (U_i, x_i) in the atlas $\mathcal{D}$ of M have the same orientation on their overlaps. There are exactly two possible orientations for an orientable manifold M. If M is an oriented manifold, the same manifold with the other possible orientation is denoted $-M$.

The boundary ∂M of an oriented manifold M inherits the orientation of M [29]. Let $p \in U$ be a boundary point of M. Then, the tangent space T_pM has a basis $\{\partial/\partial x^1, \partial/\partial x^2, \ldots, \partial/\partial x^n\}$ given by a chart (U, x). One can declare that the basis $\{\partial/\partial \bar{x}^1, \partial/\partial \bar{x}^2, \ldots, \partial/\partial \bar{x}^{n-1}\}$ of $T_p\partial M$ given by the chart $(\bar{U}, \bar{x})$ of ∂M has the same orientation as the basis $\{-\partial/\partial x^n, \partial/\partial x^1, \ldots, \partial/\partial x^{n-1}\}$ of T_pM. The tangent vector $n_p := -\partial/\partial x^n$ is in some sense an outward normal vector of the manifold-with-boundary M at a boundary point $p \in \partial M$.

Riemannian manifold

Let each tangent space T_pM have an inner product denoted by $g_p(\cdot, \cdot)$. On a coordinate chart (U, x) of M, the inner product g_p can be represented by a symmetric positive definite $n \times n$ matrix G_x. If each component function $g_{ij} : \mathbb{R}^n \to \mathbb{F}$ is smooth, the point-wise inner products g_p define a **metric tensor** g on M. A differentiable n-manifold M together with a metric tensor g on M is called a **Riemannian n-manifold** (M, g). One often abbreviates this to just M if the metric tensor is clear from the context.

The metric tensor induces an isomorphism between the tangent space T_pM and the cotangent space T_p^*M of a manifold, the flat and sharp operators. Such pairing between the tangent and cotangent spaces is symmetric positive definite. In category theory, one generalizes such pairing so that a different class of the pairing introduces a different *geometry* for the manifold. For example, for a **symplectic manifold** the pairing is nondegenerate and skew-symmetric. That is, a point-wise 2-covector whose component functions are smooth.

2.2.4 Cell complex of a manifold

A cell complex is a decomposition of a topological space into a collection of simple topological spaces, cells, that are nicely glued together along their boundaries. We can assign an orientation for each cell, from which we can construct an incidence map for pairs of cells. In this thesis, we only consider finite cell decompositions of manifolds.

In practice, the finite element mesh of a domain $\Omega \subset \mathbb{R}^n$ qualifies as the cell decomposition of the manifold M whose coordinate chart maps it to Ω. The incidence maps will be the incidence matrices of the mesh.

Let B^k denote the open 0-centered unit k-ball $B_1^k(0)$ in the space $(\mathbb{R}^n, \|\cdot\|)$. An **inner oriented k-cell** is a pair (σ, o) of piecewise diffeomorphism $\sigma^k : B^k \to \sigma^k(B^k) \subset M$ to the manifold M and orientation o for B^k. Positively oriented k-cell is abbreviated to σ^k and a negatively oriented k-cells is denoted by $-\sigma^k$. The image $\sigma_k(B^k) \subset M$ of σ^k is also denoted by σ^k, and for a k-cell σ^k we denote $\dim \sigma^k := k$.

A **geometric realization** of a k-cell is the image of the composite map $(x \circ \sigma^k)(B^k)$, where (U, x) is a chart of M so that $\sigma^k \subset U$. A k-dimensional finite element of a finite element mesh can be considered to be a geometric realization of a k-cell.

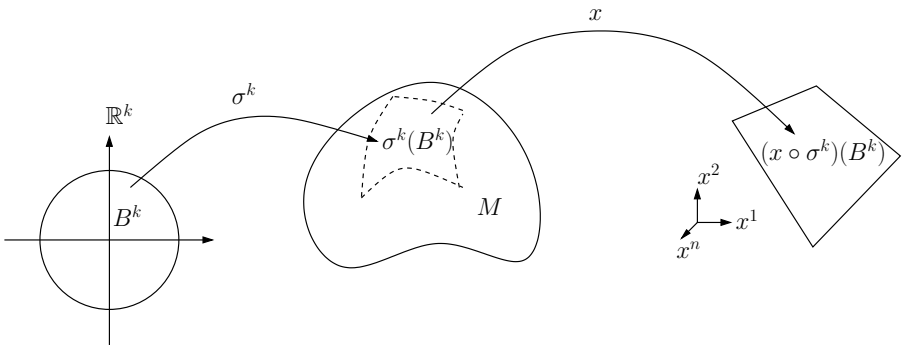


Figure 2.2: A k-cell σ^k is a local diffeomorphism from a unit k-ball B^k to a manifold M. Using a chart (U, x) it has a geometric realization.

A **cell decomposition** $(M, \mathcal{S})$ of the manifold M is a collection $\mathcal{S} = \bigcup_{k=0}^n \mathcal{S}^k$, $\mathcal{S}^k = \{\sigma_i^k\}$ of various dimensional positively oriented cells with disjoint images that satisfies

$$M = \bigcup_{k=0}^{n} \bigcup_{\sigma_i^k \in \mathcal{S}^k} \sigma_i^k(B^k). \tag{2.67}$$

The cell decomposition is required to qualify as a **regular CW-complex** [31]. However, the technical discussion of CW-complexes is omitted for two reasons: First, the full generality of the CW-complex is not needed in the computations, as discussed in [61]. Second, it is difficult to perceive the nature of the CW-complex from its definition.

However, to give a preliminary characterization for the regular CW-complexes we encounter in this thesis, the following two properties are required to hold. Also see the figure 2.3.

1. The intersection of images of two distinct cells is always empty:

$$\sigma^k \cap \sigma^l = \emptyset \quad \forall\, \sigma^k, \sigma^l \in \mathcal{S}, \quad \sigma^k \neq \sigma^l. \tag{2.68}$$

2. The intersection of closure of images of two cells is either empty or a closure of some cell image:

$$\overline{\sigma^k} \cap \overline{\sigma^l} = \begin{cases} \emptyset \\ \overline{\sigma^m} \text{ for some } \sigma^m \in \mathcal{S} \end{cases}. \tag{2.69}$$

These two properties allows one to assign a boundary for k-cells that is a union of $k-1$-cells. If $\overline{\sigma^k} \cap \overline{\sigma^l} = \overline{\sigma^l}$ holds, we call the cell σ^l a **face** of the cell σ^k. That is, when two cells "meet", their interface is a common face of both. In particular, each cell is its own face.

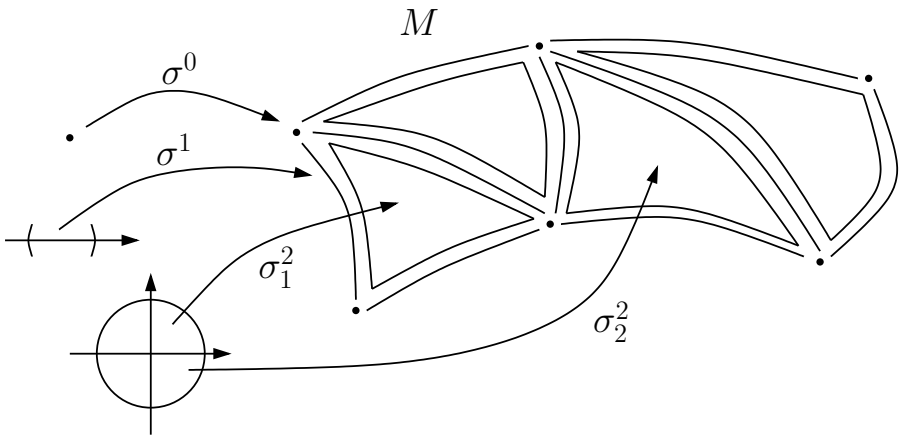


Figure 2.3: A cell decomposition that is a *regular cell complex* of a 2-manifold M.

The closure of the image $\overline{\sigma_i^k} \subset M$ of an oriented k-cell is an oriented k-manifold M_i^k with boundary ∂M_i^k that inherits the orientation. Let the oriented $k-1$-cell σ_j^{k-1} be a face of σ_i^k and M_j^{k-1} the corresponding oriented $k-1$-manifold. Then M_j^{k-1} is a submanifold of ∂M_i^k. **The orientations of the k-cell σ_i^k and a $k-1$-cell σ_j^{k-1} in the cell decomposition $\mathcal{S}$ agree**, if the oriented manifolds ∂M_i^k and M_j^{k-1} have the same orientation: One can cover ∂M_i^k with an atlas $\{(U_j, x_j)\}$ so that x_j equals the inverse of $\pm\sigma_j^{k-1}$, the sign being dominated by the inherited orientation of ∂M_i^k. The orientations agree, if the determinants of the Jacobian matrices of the transition maps $\sigma_j^{k-1} \circ x_j^{-1}$ are positive almost everywhere[4] in $x(U_j) \subset \mathbb{R}^n$.

The **incidence relation** ι of the cells in $\mathcal{S}$ is a map $\iota_k : \mathcal{S}^k \times \mathcal{S}^{k-1} \to \{0, 1, -1\}$ for each k:

$$\iota(\sigma^k, \sigma^l) := \begin{cases} 0 & \text{if } \sigma^{k-1} \text{ is not a face of } \sigma^k \\ 1 & \text{if the orientations of } \sigma^k \text{ and } \sigma^{k-1} \text{ agree} \\ -1 & \text{if the orientations of } \sigma^k \text{ and } \sigma^{k-1} \text{ do not agree} \end{cases} \tag{2.70}$$

Cell complex $K = (M, \mathcal{K}, \iota)$ of a n-manifold M is a regular CW-complex $(M, \mathcal{K})$ together with the incidence relation ι of the cells in $\mathcal{K}$. A cell complex $L = (S, \mathcal{L}, \iota)$ is a **cell subcomplex** of a cell complex $K = (S, \mathcal{K}, \iota)$ if S is a submanifold of M and $(S, \mathcal{L})$ is a *regular CW-subcomplex* of $(M, \mathcal{K})$. If the discussed cell complex K of the manifolds M is clear from the context, we write $\sigma^k \in M$ instead of $\sigma^k \in \mathcal{K}$ to indicate that the given k-cells belongs to the cell complex $K = (M, \mathcal{K}, \iota)$ of M. We also denote by $\sigma^k \in M/S$ that the k-cell belongs to the cell complex of M, but ***not*** in the cell complex L of the submanifold S of M.

[4]The map σ_j^{k-1} is only required to be a local diffeomorphism.

2.2.5 Finite elements

In the finite element method the domain $D \subset \mathbb{R}^n$ is covered with a mesh. The finite element mesh is a collection of convex polyhedral elements in $\mathbb{R}^n$ with nonempty interiors. The intersection of any two elements is required to be either empty or a common face of each of some dimension. With each element of the mesh and with each face of the elements one can associate a k-cell, and the collection of such k-cells qualifies as a regular CW-complex.

A k-dimensional **finite element** is a pair (E^k, ϕ) of **Euclidean reference k-cell** $E^k \subset \mathbb{R}^k$ and a local diffeomorphism $\phi : E^k \to \phi(E^k) \subset \mathbb{R}^n$. The Euclidean reference k-cells come in many shapes. However, usually they are *convex hulls* of a set of points, called vertices, in $\mathbb{R}^k$. The most common such Euclidean reference k-cells are *k-simplices* Δ^k. If all the finite elements are simplices, their the resulting regular CW-complex is a **simplicial complex** [31]. An oriented **standard k-simplex** Δ^k is a set

$$\Delta^k = \{(x_1 \dots, x_k) \in \mathbb{R}^k \mid \sum_{i=1}^{k} x_i \leq 1,\, x_i \geq 0 \,\forall\, i\} \subset \mathbb{R}^k \tag{2.71}$$

The $k+1$ **vertices** of a standard k-simplex are the points $0, \mathbf{e}_i \in \mathbb{R}^k$ in the standard basis of $\mathbb{R}^k$.

The finite elements are used to approximate fields on the domain. In addition to vertices, finite element may have special points in its convex hull, which together with vertices are called **nodes**. To each node $\mathbf{u}_i \in E^k$ one associates a polynomial shape function $\lambda^i : E^k \to \mathbb{R}$ which satisfies the condition $\lambda^i(\mathbf{u}_j) = \delta^i_j$.

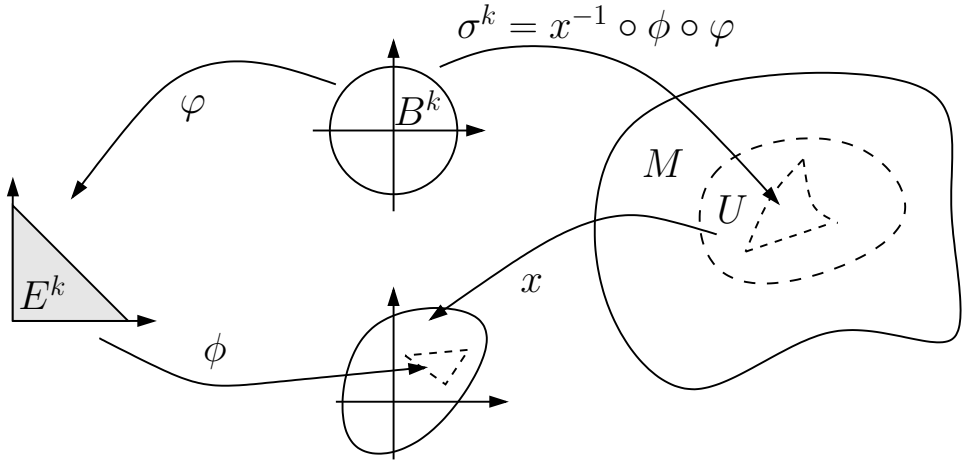


Figure 2.4: Finite element (E^k, ϕ) in $\mathbb{R}^n$ interpreted as a k-cell σ^k of an n-manifold M.

The map $\phi : E^k \to \mathbb{R}^n$ is then given by

$$\phi(\mathbf{u}) = \sum_i \lambda^i(\mathbf{u})\phi(\mathbf{u}_i), \tag{2.72}$$

where the values $\phi(\mathbf{u}_i) \in \mathbb{R}^n$ are known from the mesh data structure. The differential of

the map $\phi = (\phi_1, \phi_2, \dots, \phi_n)$ is given by Jacobian matrix J at point $\mathbf{u} = (u_1, u_2, \dots u_k)$:

$$\mathrm{J} := \begin{bmatrix} \frac{\partial \phi_1}{\partial u^1} & \cdots & \frac{\partial \phi_1}{\partial u^k} \\ \vdots & \ddots & \vdots \\ \frac{\partial \phi_n}{\partial u^1} & \cdots & \frac{\partial \phi_n}{\partial u^k} \end{bmatrix} = \sum_i \begin{bmatrix} \frac{\partial \lambda^i}{\partial u^1} x_i^1 & \cdots & \frac{\partial \lambda^i}{\partial u^k} x_i^1 \\ \vdots & \ddots & \vdots \\ \frac{\partial \lambda^i}{\partial u^1} x_i^n & \cdots & \frac{\partial \lambda^i}{\partial u^k} x_i^n \end{bmatrix} \quad \text{where } \phi(\mathbf{u}_i) = (x_i^1, x_i^2, \dots x_i^n) \in \mathbb{R}^n \tag{2.73}$$

A k-dimensional finite element (E^k, ϕ) can be interpreted as a k-cell $\sigma^k \in M$ when one insists that ϕ maps E^k to a chart (U, x) of an n-manifold M, see figure 2.4. That is, if one considers a piecewise diffeomorphism $\varphi : B^k \to E^k$, the corresponding k-cell map is $\sigma^k = x^{-1} \circ \phi \circ \varphi : B^k \to M$ as required. Later in this thesis, finite elements are also denoted by $\sigma^k = (E^k, \phi)$, since they can be interpreted as k-cells.

2.3 Homology and cohomology of a manifold

While homology and cohomology can be considered to be purely algebraic concepts, a cell decomposition of a manifold (or more generally of a topological space) can be used to construct algebraic chain complexes.

Such chain complexes have well defined homology and cohomology groups, and their homological properties reflect some topological properties of the manifold. Those properties we call the homology and the cohomology of the manifold, as they are independent of the cell decomposition we chose.

If the cells in a cell complex of a manifold are given a geometric realization, so are the chains as the formal sums of cells. Since homology and cohomology groups consists of cosets of chains and cochains, they can also be given a geometric realization by realizing a representative element. That is, we are able to visually express the homology and the cohomology of the manifold.

Later in this thesis our goal is to formulate electromagnetic field problems on a manifold. In the following sections we show that such fields can be regarded as cochains. Therefore, the homology and the cohomology of the manifold and the cohomology of the fields are intimately linked, and need to be taken into account in the formulation of the field problems.

2.3.1 Chain complexes of a manifold

Given cell complexes $K = (M, \mathcal{K}, \iota)$ and $L = (S, \mathcal{L}, \iota)$ of a manifold M and its submanifold S, we construct finitely generated abelian groups $C_k(M)$, $C_k(S)$, and $C_k(M, S)$ as formal sums of k-cells σ_i^k in M, S, and $M \setminus S$, respectively.

Using the incidence relation ι of cells, we construct the boundary homomorphisms ∂ between the abelian groups. As a result, we obtain chain complexes $\mathcal{C}(M) = (C_\bullet(M), \partial_\bullet)$ and $\mathcal{C}(S) = (C_\bullet(S), \partial_\bullet)$, and a relative chain complex $\mathcal{C}(M, S) = (C_\bullet(M, S), \partial_\bullet)$.

Construction of the abelian groups

The (relative) chain complexes $\mathcal{C}(M)$, $\mathcal{C}(S)$, and $\mathcal{C}(M,S)$ of are constructed as follows. A k-**chain** $c \in C_k(M)$ is a formal sum of positively oriented k-cells σ^k of M:

$$c := \sum_{\sigma_i^k \in M} c^i \sigma_i^k = \sum_{\sigma_i^k \in M} -c^i(-\sigma_i^k) = (\sigma_1^k, \sigma_2^k, \ldots, \sigma_{q_k(M)}^k)\mathbf{c} = \Sigma_k \mathbf{c} \tag{2.74}$$

holds, where $q_k(M)$ is the number of k-cells in M and the integer coefficient vector $\mathbf{c} \in \mathbb{Z}^{q_k(M)}$ represents the k-chain. That is, the k-cells in $\mathcal{S}^k$ form a **cell basis** $\Sigma_k = (\sigma_1^k, \sigma_2^k, \ldots, \sigma_{q_k(M)}^k)$ for the group $C_k(M)$. Thus, a k-cell is treated as a simple k-chain. A k-chain $c \in C_k(S)$ is a similar formal sum of cells in S, which constitute a basis of the group $C_k(S)$.

To define the **relative** k-**chains** $c_k \in C_k(M,S)$, a surjective homomorphism $j : C_k(M) \to C_k(M,S)$ needs to be defined. To make the group $C_k(M,S)$ free abelian, i.e. to have no torsion subgroups, the following definition is chosen:

$$j(\sigma^k) = \begin{cases} \sigma^k & \text{if } \sigma^k \notin S \\ 0 & \text{otherwise} \end{cases} . \tag{2.75}$$

holds. Then, a relative k-chain $c \in C_k(M,S)$ is defined to be a formal sum of k-cells in $M \setminus S$:

$$c := \sum_{\sigma_i^k \in M \setminus S} \mathbf{c}^i \sigma_i^k = (\sigma_1^k, \sigma_2^k, \ldots, \sigma_{q_k(M\setminus S)}^k)\mathbf{c} = \Sigma_k \mathbf{c} \tag{2.76}$$

That is, the basis of $C_k(M,S)$ is formed by the k-cells in $M \setminus S$.

Construction of boundary homomorphisms

Let $K = (M, \mathcal{S}, \iota)$ be a cell complex of M. From the incidence relation ι of the cells one can construct incidence integer $|\mathcal{S}^{k-1}| \times |\mathcal{S}^k|$ matrices D_k for each k. The matrix can be arranged into blocks

$$\mathrm{D}_k = \begin{bmatrix} \bar{\mathrm{D}}_k & \mathrm{A}_k \\ \mathrm{B}_k & \acute{\mathrm{D}}_k \end{bmatrix}, \tag{2.77}$$

where the matrices $\bar{\mathrm{D}}_k$ and $\acute{\mathrm{D}}_k$ contain the incidence information of cells in $M \setminus S$ and S, respectively. The matrices A_k and B_k decode the incidence for pairs of cells where other cell is in $M \setminus S$ and the other is in S.

Then, the boundary homomorphisms $\partial : C_k(M) \to C_{k-1}(M)$, $\acute{\partial} : C_k(S) \to C_{k-1}(S)$, and $\bar{\partial} : C_k(M,S) \to C_{k-1}(M,S)$ evaluate as

$$\partial c = \sum_{\sigma_i^{k-1} \in M} (\mathrm{D}_k \mathbf{c})^i \sigma_i^{k-1} = \Sigma_{k-1} \mathrm{D}_k \mathbf{c} \quad \forall\, c \in C_k(M), \tag{2.78}$$

$$\acute{\partial} c = \sum_{\sigma_i^{k-1} \in S} (\acute{\mathrm{D}}_k \mathbf{c})^i \sigma_i^{k-1} = \Sigma_{k-1} \acute{\mathrm{D}}_k \mathbf{c} \quad \forall\, c \in C_k(S), \text{ and} \tag{2.79}$$

$$\bar{\partial} c = \sum_{\sigma_i^{k-1} \in M \setminus S} (\bar{\mathrm{D}}_k \mathbf{c})^i \sigma_i^{k-1} = \Sigma_{k-1} \bar{\mathrm{D}}_k \mathbf{c} \quad \forall\, c \in C_k(M,S). \tag{2.80}$$

Later, we will denote all of them by plain ∂, since there's usually no room for ambiguity.

Inner product of chains

Free abelian groups $C_k(M)$ are endowed with a "inner product". With the cell basis, this allows us to weight k-chains how many k-cells they occupy and on how many k-cells two k-chains overlap. The inner product $\langle \cdot, \cdot \rangle : C_k(M) \times C_k(M) \to \mathbb{Z}$ is defined by

$$\langle a, b \rangle = \langle \sum_{\sigma_i^k \in M} a^i \sigma_i^k, \sum_{\sigma_j^k \in M} b^j \sigma_j^k \rangle := \sum_{\sigma_i^k \in M} \sum_{\sigma_j^k \in M} a^i b^j \iota(\sigma_i^k, \sigma_j^k) = \mathbf{a} \cdot \mathbf{b} \quad \forall\, a, b \in C_k(M). \tag{2.81}$$

That is, for the k-cells $\langle \sigma_i^k, \sigma_j^k \rangle = \iota(\sigma_i^k, \sigma_j^k) = \delta_{ij}$ holds. A $k-1$-chain c_2 is said to be on the boundary of an k-chain c_1, if $\langle \partial c_1, c_2 \rangle \neq 0$ holds. Then, c_1 is said to be on the coboundary of c_2.

2.3.2 Cochain complexes of a manifold

Let $\{\epsilon_k^i\}$ be such set of k-cochains that $\epsilon^i(\sigma_j^k) = \delta_j^i$ hold. They form a basis $\Sigma^k = (\epsilon_k^1, \epsilon_k^2, \ldots, \epsilon_k^{q_k(M)})$ for the finitely generated free abelian group $C^k(M) := C^k(\mathcal{C}(M); G)$. Then, a k-cochain γ has the representation

$$\gamma = \sum_{\sigma_i^k \in M} \gamma_i \epsilon^i = (\epsilon_k^1, \epsilon_k^2, \ldots, \epsilon_k^{q_k(M)}) \boldsymbol{\gamma} = \Sigma^k \boldsymbol{\gamma} \tag{2.82}$$

where the elements γ_i of the coefficient vector $\boldsymbol{\gamma} \in G^{q_k(M)}$ are in the abelian group G. The k-cochain group $C^k(S)$ has a similar basis in terms of k-cells in S. To define the relative k-cochain group $C^k(M, S)$, the injective map $\tilde{j} : C^k(M, S) \to C^k(M)$ is chosen to be such that $\tilde{j}(\gamma) = \gamma \circ j$ holds for all $\gamma \in C^k(M, S)$. This definition ensures that the bilinear pairing $C^k(M, S) \times C_k(M, S) \to G$ given by $\gamma(c) = \boldsymbol{\gamma}^T \mathbf{c}$ is nondegenerate. That is, the following hold:

$$\gamma(c) = 0 \quad \forall\, \gamma \in C^k(M, S) \implies c \in C_k(S) \tag{2.83}$$

$$\gamma(c) = 0 \quad \forall\, c \in C_k(M, S) \implies \gamma = 0. \tag{2.84}$$

Since $\delta_k \gamma = \gamma \circ \partial_{k+1}$ holds by definition (2.28), from $(\delta_k \gamma)(c) = \gamma(\partial_{k+1} c) = \boldsymbol{\gamma}^T \mathrm{D}_{k+1} \mathbf{c} = (\mathrm{D}_{k+1}^T \boldsymbol{\gamma})^T \mathbf{c}$ one can conclude that for $\gamma = \Sigma^k \boldsymbol{\gamma} \in C^k(M)$

$$\delta_k \gamma = \Sigma^{k+1} \mathrm{D}_{k+1}^T \boldsymbol{\gamma} \tag{2.85}$$

holds, i.e. the coboundary homomorphism δ_k is represented by the transpose of the integer matrix D_{k+1} that represents the boundary homomorphism ∂_{k+1}. Therefore, the coboundary homomorphisms $\acute{\delta}_k$ and $\bar{\delta}_k$ of $C^k(S)$ and $C^k(M, S)$ are represented by the blocks $\acute{\mathrm{D}}_{k+1}^{\mathrm{T}}$ and $\bar{\mathrm{D}}_{k+1}^T$ of D_{k+1}^T, respectively.

2.3.3 Homology and cohomology of a manifold

Given two cell decompositions K, K' of a manifold M and corresponding cell decompositions L, L' of its submanifold S, it can be shown that the homology and the cohomology groups induced by the cell complexes K and L are isomorphic to those induced by K' and

L' [31, 61]. That is, ***any cell complexes of the manifold*** M ***and its submanifold*** S ***can be used to determine the (relative) homology and cohomology groups of*** M ***and*** S***.***

We denote by $H_k(M) := H_k(\mathcal{C}(M))$ and $H^k(M;G) := H^k(\mathcal{C}(M);G)$ the homology and cohomology groups of the manifold M covered with any cell complex K. Similarly, relative homology and cohomology groups are denoted by $H_k(M,S) := H_k(\mathcal{C}(M,S))$ and $H^k(M,S;G) := H^k(\mathcal{C}(M,S);G)$. The relative homology and cohomology groups $H_k(M,S)$ and $H^k(M,S;G)$ are also called k:th homology and k:th cohomology group of M modulo S, respectively.

Homotopic manifolds and homology

In topology, there are numerous ways to define in what respect two topological spaces are the same. That is, topological spaces can be categorized to equivalence classes according to many different criteria. The strongest of such notions is to state that the topological spaces X and Y are equivalent if there exists a homeomorphism $f : X \to Y$. For example, the images of any two k-cells are equivalent in that sense, while an n-ball $X = B_1^n(0)$ and a single point $Y = \{0\}$ in $(\mathbb{R}^n, \tau)$ would fall in different equivalence classes for $n > 0$.

If two manifolds M and M' are homeomorphic, they have isomorphic homology groups. For then one can construct a cell complex $K' = (M', \mathcal{K}', \iota')$ where each cell is an image of a unique cell in the cell complex $K = (M, \mathcal{K}, \iota)$. Consequently, $\iota = \iota'$ holds and the constructed chain complexes are indistinguishable.

However, from the point of view of homology, the notion that homeomorphic spaces are equivalent is too strong, since a certain strictly larger class of continuous maps between two manifolds will suffice to imply isomorphic homology groups for the manifolds. Such maps are called *homotopy equivalences* [31].

A **homotopy** is a family of maps $f_t : X \to Y$, $t \in [0,1] \subset \mathbb{R}$, such that the associated map $(x,t) \to f_t(x) : X \times [0,1] \to Y$ is continuous. The maps f_0 and f_1 are **homotopic**, if there exists a homotopy f_t that connects them. A map $f : X \to Y$ is called a **homotopy equivalence** if there exists a map $g : Y \to X$ such that the maps $g \circ f$ and $f \circ g$ are homotopic to the identity maps of X and Y, respectively. Then the spaces X and Y are said to have the same homotopy type.

Note the similarities with the concept of chain homotopy. Indeed, if $f : M \to M'$ is a homotopy equivalence between manifolds, then the induced maps $\phi_\bullet(f) : C_\bullet(M) \to C_\bullet(M')$ are chain equivalences [31]. Thus, ***the chain complexes constructed from manifolds with the same homotopy type have isomorphic homology groups.***

Remark 2.3.1. However, a chain equivalence does not necessarily induce a homotopy equivalence.

Duality of homology and cohomology on a manifold

In section 2.1.3 on page 25, we mentioned that the cohomology vector space $H^k(\mathcal{C};\mathbb{F})$ is the dual space of the homology vector space $H_k(\mathcal{C};\mathbb{F})$. That is, the homology and cohomology spaces are isomorphic. On a closed orientable manifold, there's a famous duality isomorphism called *Poincarè duality*, which generalizes to a compact manifold-with-boundary.

Let M be a compact orientable n-manifold and let $\partial M = S_1 \cup S_2$ hold, where S_1 and S_2 are its $n-1$-submanifolds with common topological boundary, i.e. $\partial S_1 = \partial S_2 = S_1 \cap S_2$ hold. Then [31]

$$H^k(M, S_1; \mathbb{F}) \simeq H_{n-k}(M, S_2; \mathbb{F}) \tag{2.86}$$

holds. In words, the k:th cohomology space modulo S_1 is isomorphic to the $n-k$:th homology space modulo S_2. If one considers a closed manifold, i.e. if $\partial M = S_1 = S_2 = \emptyset$ holds, one recovers the **Poincarè duality**. Another special case of (2.86), when $S_1 = \partial M$ and $S_2 = \emptyset$ hold, is called the **Lefschetz duality**.

The isomorphism (2.86) has tangible interpretations in computational electromagnetics that are related to the circuit coupling of an electromagnetic boundary value problem. This will be discussed in chapter 4.

Long exact sequences of homology and cohomology

Various homology groups of a manifold are related by two long exact sequences. They can be applied to deduce isomorphisms between homology groups if some of the involved homology groups are trivial. This is often the case with manifolds that are used in finite element modeling. The long exact sequences of homology also have their cohomological counterparts.

Let M be an n-manifold and S its submanifold. **The long exact homology sequence** is

$$0 \to H_n(S) \to \ldots \to H_k(S) \xrightarrow{i_*} H_k(M) \xrightarrow{j_*} H_k(M, S) \xrightarrow{\partial_*} H_{k-1}(S) \to \ldots \to H_0(M, S) \to 0. \tag{2.87}$$

It is induced by the short exact sequences

$$0 \to C_k(S) \xrightarrow{i} C_k(M) \xrightarrow{j} C_k(M, S) \to 0 \tag{2.88}$$

of chain complexes together with the *snake lemma* [48, 31], which provides the connecting homomorphisms $\partial_* : H_k(M, S) \to H_{k-1}(S)$ for each k.

Let M be a manifold and S its submanifold. Also let M_1 and M_2 be submanifolds of M so that $M = M_1^\circ \cup M_2^\circ$ holds, i.e. M is the union of the interiors of M_1 and M_2, and let $S_1 \subset M_1$ and $S_2 \subset M_2$ be submanifolds of S so that $S = S_1^\circ \cup S_2^\circ$ holds. The relative **Mayer-Vietoris sequence** of homology is [31]

$$\begin{aligned} \ldots \to & H_k(M_1 \cap M_2, S_1 \cap S_2) \to H_k(M_1, S_1) \oplus H_k(M_2, S_2) \to H_k(M, S) \to \\ & H_{k-1}(M_1 \cap M_2, S_1 \cap S_2) \to \ldots \end{aligned} \tag{2.89}$$

The non-relative version of this sequence is obtained by setting $S = S_1 = S_2 = \emptyset$.

The long exact homology sequence and the Mayer-Vietoris sequence have their cohomological counterparts, where the directions of the maps are reversed.

2.4 Differential forms

Differential forms describe integrands on manifolds in a coordinate-free manner. That is, they are apt to model various *densities* in physics. As a consequence to the coordinate-free construction, the change of variables formula of integration is a built-in property in the integration of differential forms.

Since the integration domain is a geometric entity of certain dimension, the degree of a differential form reflects this dimension. Also, the exterior algebra structure of differential forms supports their geometric nature. It also reflects the Fubini's theorem of integration which states that one can construct multiple integrals by iterated integration.

The exterior derivative of differential forms is also constructed with powerful classical results in mind. Namely, the Stokes', Green's, and Divergence theorems, which are with differential forms combined into a single theorem: the generalized Stokes theorem.

It turns out that differential forms can be regarded as cochains. This observation gives cohomology a role in analysis on manifolds. A cohomology theory whose k-cochains are differential k-forms is called de Rham cohomology. Integration domains on a manifold can be given the structure of a chain complex. Since integration maps the integration domain to a scalar value, and by the generalized Stokes' theorem, differential forms together with their exterior derivative form a cochain complex.

Lastly, the study of harmonic differential forms is the modern theory of irrotational and solenoidal vector fields, which is a prominent concept in physics. Harmonic differential forms are in close relationship with the de Rham cohomology. Therefore, homology and cohomology computation can be utilized in the computation of harmonic differential forms.

2.4.1 The basic construction

On a differentiable manifold M, a differential k-form ω smoothly assigns to a point $p \in M$ a k-covector ω_p. Therefore in a sense, differential forms are a generalization of vector fields in the classical vector analysis.

The k-covector ω_p is an element of $\Lambda^k(T_p^*)$. Since the cotangent space T_p^* is different for each $p \in M$, the *tangent bundle* needs to be introduced in the formal definition of differential forms to make them an apt generalization of vector fields. In contrast, classical vector fields have their values in the same vector space, $\mathbb{R}^3$.

The **tangent bundle** TM of a differentiable manifold is defined as a set

$$TM := \bigcup_{p \in M} \{(p, v_p) \,|\, p \in M, v_p \in T_pM\} \tag{2.90}$$

together with a surjective projection $\pi : TM \to M$ defined as $\pi(p, v_p) := p$. That is, the elements of the tangent bundle are pairs of points and tangent vectors on the manifold, and the projection map picks the point from the pair[5]. The **cotangent bundle** T^*M is defined similarly. Analogously, the **k-cotangent bundle** defined as the set

$$\Lambda^k(T^*M) := \bigcup_{p \in M} \{(p, \omega_p) \,|\, p \in M, \omega_p \in \Lambda^k(T_p^*M)\}. \tag{2.91}$$

[5]Interestingly, tangent bundle TM is always a differentiable manifold of its own right [24], and has its own tangent bundle TTM.

A differential k-form ω is a smooth map $M \to \Lambda^k(T^*M)$ that satisfies $\pi \circ \omega = \mathrm{id}$. This property ensures that to each point $p \in M$ a k-covector of its own k-cotangent space $\Lambda^k(T_p^*M)$ is assigned. A differential 0-form is simply a smooth map $M \to \mathbb{F}$.

Given a coordinate chart (U, x) for M, a k-form ω can be represented in the coordinate cobasis $\{\mathrm{d}x^1, \mathrm{d}x^2, \ldots, \mathrm{d}x^n\}$ of each T_p^*M as

$$\omega = \sum_{\vec{I}} \omega_{\vec{I}} \mathrm{d}x^{\vec{I}}, \tag{2.92}$$

where the maps $\omega_{\vec{I}} : M \to \mathbb{F}$ are smooth, i.e. 0-forms.

The set of all k-forms on M constitutes a vector space, denoted $\Omega^k(M)$, with the vector space operations defined such that in each $p \in M$

$$(\omega + \eta)_p(v) = \omega_p(v) + \eta_p(v) \tag{2.93}$$

$$(f\omega)_p(v) = f(p)\omega_p(v) \tag{2.94}$$

hold for all smooth maps $f : M \to \mathbb{F}$ and for all $v \in \Lambda^k(T_pM)$. The exterior algebra structure and the Hodge operator of differential forms is defined in a similar point-wise manner. The exterior product of a k-form ω and an l-form η is defined as $(\omega \wedge \eta)_p := \omega_p \wedge \eta_p$. On an orientable Riemannian n-manifold (M, g), the Hodge operator is defined as $(\star\omega)_p = \star_p \omega_p$, where $\star_p$ is defined via the inner product g_p of the tangent space T_pM as in the equation (2.52).

Let $\phi : M \to N$ be a differentiable map from an n-manifold M to m-manifold N. Let $p \in M$ and (U, x) be a chart for M and (V, y) be a chart for N. For $p \in U$ and $\phi(p) \in V$, the matrix J of partial derivatives

$$\mathrm{J}_i^j = \left.\frac{\partial(y^j \circ \phi \circ x^{-1})}{\partial x^i}\right|_{x(p)} \tag{2.95}$$

is an $m \times n$ matrix. That is, J is a representation for the linear map $\mathrm{d}\phi : T_pM \to T_{\phi(p)}N$ called the **differential** of ϕ. Thus, $\mathrm{d}\phi$ induces a pullback $\phi^* : \Omega^k(N) \to \Omega^k(M)$ of differential k-forms with the point-wise definition

$$(\phi^*\omega)_p := \phi_p^* \omega_p. \tag{2.96}$$

If S is a submanifold of M, the pullback by the inclusion map $i : S \to M$ is called **trace** and denoted by $\mathrm{t}_S\omega := i^*\omega$. If S is clear from the context, we omit it.

The exterior derivative of differential k-forms is a linear map $\mathrm{d} : \Omega^k(M) \to \Omega^{k+1}(M)$. Using the chart representations, for $\omega = \sum_{\vec{I}} \left(\omega_{\vec{I}} \mathrm{d}x^{\vec{I}}\right)$ it is defined by

$$\mathrm{d}\omega := \sum_{\vec{I}} \sum_{i=1}^{n} \frac{\partial \omega_{\vec{I}}}{\partial x^i} \mathrm{d}x^i \wedge \mathrm{d}x^{\vec{I}}. \tag{2.97}$$

A k-form ω is called **closed**, if $\mathrm{d}\omega = 0$ holds, and **exact**, if there exists a $k-1$-form η such that $\omega = \mathrm{d}\eta$ holds. Importantly, the exterior derivative obeys the **Leibniz product rule**:

$$\mathrm{d}(\omega \wedge \eta) = \mathrm{d}\omega \wedge \eta + (-1)^k \omega \wedge \mathrm{d}\eta, \tag{2.98}$$

where ω is a k-form and η is an l-form. The exterior derivative also commutes with the pullback, and therefore with the trace operator. This property is put into good use in the analysis of boundary value problems.

2.4.2 Integration

Differential k-forms are tailored to be integrated over k-chains. That is, over linear combinations of k-cells on an n-manifold M. Let $\mathcal{C}(M)$ be a chain complex of M with bases $(\sigma_1^k, \sigma_2^k, \ldots, \sigma_{q_k(M)}^k)$ for each $C_k(M)$. Since a k-chain $c \in C_k(M)$ is a linear combination of the basis k-cells, we define

$$\int_c \omega = \int_{\sum_i c^i \sigma_i^k} \omega := \sum_i c^i \int_{\sigma_i^k} \omega \tag{2.99}$$

The integration of the k-form ω over a k-cell σ_i^k is defined by using the pullback $(\sigma_i^k)^*$. Once ω is pulled back to B^k, one can use *Lebesgue integration*[6] of $\mathbb{R}^k$ to integrate the component functions $\omega_{\vec{I}}$ of ω:

$$\int_{\sigma_i^k} \omega := \int_{B^k} (\sigma_i^k)^* \omega = \int_{B^k} \omega_{12\ldots k} \mathrm{d}u^1 \wedge \mathrm{d}u^2 \wedge \ldots \wedge \mathrm{d}u^k \tag{2.100}$$

where $\{\mathrm{d}u^1, \mathrm{d}u^2, \ldots, \mathrm{d}u^k\}$ is the coordinate cobasis of the unit ball B^k and $\mathrm{d}u^1 \wedge \mathrm{d}u^2 \wedge \ldots \wedge \mathrm{d}u^k$ is interpreted as the *Lebesgue measure* [25] of $\mathbb{R}^k$.

If $\phi : M \to N$ is a differentiable map between manifolds, the **change of variables formula** for differential k-forms reads:

$$\int_c \phi^* \omega = \int_{\phi_*(c)} \omega := \sum_i c^i \int_{\phi \circ \sigma_i^k} \omega, \tag{2.101}$$

where ϕ_* is a chain map induced by ϕ and defined via the k-cells by $\phi_*(\sigma_i^k) = \phi \circ \sigma_i^k$, which is is a k-cell of the manifold N. The **generalized Stokes' theorem** is

$$\int_c \mathrm{d}\omega = \int_{\partial c} \omega. \tag{2.102}$$

As differential forms are a special case of cochains, the generalized Stokes' theorem could also be taken as the definition of the exterior derivative instead of the coordinate chart definition (2.97). That is, in the sense of the definition of the coboundary homomorphism (2.28) on the page 23.

Together with the product rule of the exterior derivative, generalized Stokes' theorem implies the following **integration by parts** formula for differential forms:

$$\int_c \mathrm{d}\omega \wedge \eta = -(-1)^k \int_c \omega \wedge \mathrm{d}\eta + \int_{\partial c} \omega \wedge \eta. \tag{2.103}$$

Integration of differential k-forms is also defined over an orientable k-manifold S. One is to construct a k-chain that consist of all k-cells of S with matching orientations and simply define $\int_S \omega := \int_c \omega$. For such k-chain c, its coset $[c]$ belongs to the relative homology space $H_k(S, \partial S)$.

[6] For the defined smooth differential forms, *Riemann integration* would suffice, but later on, we will demand only piecewise smoothness.

2.4.3 de Rham cohomology

The integration of differential k-form ω on an n-manifold M is a linear map $\int_{(\cdot)}\omega : \Omega^k(M) \to \mathbb{F}$. Since their exterior derivative satisfies the generalized Stokes' theorem, differential forms constitute the cochain complex

$$0 \to \Omega^0(M) \xrightarrow{\mathrm{d}} \ldots \xrightarrow{\mathrm{d}} \Omega^k(M) \xrightarrow{\mathrm{d}} \Omega^{k+1}(M) \xrightarrow{\mathrm{d}} \ldots \xrightarrow{\mathrm{d}} \Omega^n(M) \to 0. \tag{2.104}$$

By construction of the section 2.1.2 on page 21, one obtains de Rham k-cohomology spaces

$$H^k_{\mathrm{dR}}(M) := \ker \mathrm{d}/\operatorname{im}\mathrm{d} = \{\omega + \mathrm{d}\eta \in \Omega^k(M) \,|\, \mathrm{d}\omega = 0,\, \eta \in \Omega^{k-1}(M)\}. \tag{2.105}$$

Let

$$\Omega^k(M,S) := \{\omega \in \Omega^k(M) \,|\, \mathrm{t}_S\omega = 0\} \tag{2.106}$$

denote the space of k-forms on M whose trace vanishes on a part of the domain $S \subset M$[7]. One has a short exact sequence

$$0 \to \Omega^k(M,S) \xrightarrow{i} \Omega^k(M) \xrightarrow{\mathrm{t}} \Omega^k(S) \to 0, \tag{2.107}$$

where i is an inclusion map. Since $\mathrm{t}\mathrm{d}\omega = \mathrm{d}\mathrm{t}\omega$ holds, one can define an operator $\bar{\mathrm{d}} : \Omega^k(M,S) \to \Omega^{k+1}(M,S)$ by $\bar{\mathrm{d}}\omega := \mathrm{d}i\omega$. Then, one has $\mathrm{t}\bar{\mathrm{d}}\omega = \mathrm{t}\mathrm{d}i\omega = \mathrm{d}\mathrm{t}i\omega = 0$ as required. Relative de Rham k-cohomology spaces are defined by

$$H^k_{\mathrm{dR}}(M,S) := \ker \bar{\mathrm{d}}/\operatorname{im}\bar{\mathrm{d}} = \{\omega + \bar{\mathrm{d}}\eta \in \Omega^k(M,S) \,|\, \bar{\mathrm{d}}\omega = 0,\, \eta \in \Omega^{k-1}(M,S)\}. \tag{2.108}$$

Later in this thesis, the operator $\bar{\mathrm{d}}$ is denoted by plain d, since there's usually no room for ambiguity.

What makes de Rham cohomology interesting is that the de Rham cohomology spaces $H^k_{\mathrm{dR}}(M,S)$ are isomorphic to the homology spaces $H_k(M,S)$. That is,

$$H^k_{\mathrm{dR}}(M,S) \simeq H_k(M,S) \tag{2.109}$$

is called relative **de Rham's theorem** [14, 28, 27, 23, 24]. In words, de Rham's theorem states that if one assigns a number $C_i \in \mathbb{F}$ for each element $[z_i]$ in a basis of $H_k(M,S)$, there exists a closed differential k-form $\omega \in \Omega^k(M,S)$ such that

$$\int_{z_i} \omega = C_i \quad \forall\, i :\, 1 \le i \le \beta_k(M,S) \tag{2.110}$$

holds. That is, for such non-zero k-form $[\omega] \in H^k_{\mathrm{dR}}(M,S)$ holds.

[7] Usually in our boundary value problem applications $S \subset \partial M$ holds.

2.4.4 Harmonic differential forms

In classical vector analysis by *Helmholtz decomposition* [32], any smooth vector field on $\mathbb{E}^3$ that vanishes at infinity is uniquely determined by its divergence and curl. In boundary value problems however, the domain manifold is often quite unlike $\mathbb{E}^3$. The manifold is compact, and may be closed, or the unknown field doesn't necessarily vanish at the boundary.

In such setting, the *Hodge decomposition* generalizes the Helmholtz decomposition. In addition to the "rotational" and "solenoidal" components, a differential k-form on an n-manifold may have so-called harmonic component. Strikingly, the harmonic component is related to the k:th de Rham cohomology space of the manifold, and therefore related to the k:th homology space of the manifold.

Let M be an n-manifold whose boundary is decomposed in two parts: $S \in \partial M$ and $S^* := \partial M / S$. It may hold that $S = \emptyset$ or $S = \partial M$. Also, the manifold M may be closed, i.e. $\partial M = S = S^* = \emptyset$ may hold.

To be able to define such decomposition for differential that are only piecewise smooth, i.e. smooth almost everywhere on M, we begin by defining so-called L^2 and *Sobolev* spaces of differential k-forms.

The space $\Omega^k(M)$ of smooth differential k-forms can be turned into an inner product space by defining the following inner product on $\Omega^k(M)$:

$$\langle \omega, \eta \rangle := \int_M \omega \wedge \star\eta. \tag{2.111}$$

Now, the set of *piecewise* smooth differential k-forms ω that satisfy $\langle \omega, \omega \rangle < \infty$ constitute the **inner product space** $L^2\Omega^k(M)$. In $L^2\Omega^k(M)$, the exterior derivative is no longer defined everywhere, so we replace it by the **weak exterior derivative**. The weak exterior derivative $\tilde{\mathrm{d}}\omega$ of $\omega \in L^2\Omega^k(M)$ is such that it satisfies

$$\int_M \tilde{\mathrm{d}}\omega \wedge \varphi = (-1)^k \int_M \omega \wedge \mathrm{d}\varphi \quad \forall\, \varphi \in \Omega^{n-k-1}(M, \partial M). \tag{2.112}$$

That is, the $(n-k-1)$-forms φ are smooth everywhere in M, and are required to have compact support and the traces $\mathrm{t}\varphi$ vanish on the boundary ∂M. If ω is everywhere smooth on M, $\tilde{\mathrm{d}}$ and d coincide. Later in this thesis, we will denote the weak exterior derivative also by d.

Next, a couple of **Sobolev spaces of differential k-forms** are defined:

$$H\Omega^k(M, S) := \{\omega \in L^2\Omega^k(M) \,|\, \mathrm{d}\omega \in L^2\Omega^{k+1}(M),\ \mathrm{t}_S\omega = 0\}, \tag{2.113}$$

$$H\Omega^k(M, S^*) := \{\omega \in L^2\Omega^k(M) \,|\, \mathrm{d}\star\omega \in L^2\Omega^{n-k-1}(M),\ \mathrm{t}_{S^*}\star\omega = 0\}. \tag{2.114}$$

The space of **harmonic differential k-forms** is defined as:

$$\mathcal{H}^k(M, S) := \{\omega \in L^2\Omega^k(M) \,|\, \mathrm{d}\omega = 0,\ \mathrm{d}\star\omega = 0,\ \mathrm{t}_S\omega = 0,\ \mathrm{t}_{S^*}\star\omega = 0\}. \tag{2.115}$$

One observes that the Hodge operator acts as an isomorphism $\star : \mathcal{H}^k(M, S) \to \mathcal{H}^{n-k}(M, S^*)$, since $\star\star\omega = (-1)^{k(n-k)}\omega$ holds for any $\omega \in \Omega^k(M)$. Note the similarities with the equation (2.86) on page 42.

An important theorem states that the space $\mathcal{H}^k(M,S)$ is finite-dimensional and that any k-form $\omega \in L^2\Omega^k(M)$ has the following orthogonal **Hodge decomposition** [57, 28, 27]

$$\omega = \mathrm{d}\alpha + (-1)^{n(k+1)+1} \star \mathrm{d} \star \beta + \gamma, \tag{2.116}$$

where $\alpha \in H\Omega^{k-1}(M,S)$, $\beta \in H\Omega^{k+1}(M,S^*)$, and $\gamma \in \mathcal{H}^k(M,S)$. The operator $(-1)^{n(k+1)+1} \star \mathrm{d} \star : \Omega^k(M) \to \Omega^{k-1}(M)$ is often called **codifferential**.

Another important theorem states that in each de Rham cohomology class $[\omega]$ of $H^k_{\mathrm{dR}}(M,S)$, there exists an unique harmonic representative k-form $\omega \in \mathcal{H}^k(M,S)$. That is,

$$H^k_{\mathrm{dR}}(M,S) \simeq \mathcal{H}^k(M,S) \tag{2.117}$$

is called relative **Hodge's theorem** [57, 28, 27, 24]. An isomorphism can be constructed as follows. Given $\omega \in \mathcal{H}^k(M,S)$, $[\omega]$ is simply represented by $\omega + \mathrm{d}\alpha$ for any $\alpha \in H\Omega^{k-1}(M,S)$. Given $[\zeta] \in H^k_{\mathrm{dR}}(M,S)$, one needs to solve the following boundary value problem for α. For any representative ζ, one seeks for such α that $\omega = \zeta + \mathrm{d}\alpha \in \mathcal{H}^k(M,S)$ holds. Then,

$$0 = \mathrm{d} \star \omega = \mathrm{d} \star \zeta + \mathrm{d} \star \mathrm{d}\alpha \tag{2.118}$$

holds for an $\alpha \in H\Omega^{k-1}(M,S)$ with boundary conditions

$$\mathrm{t}_S \mathrm{d}\alpha = -\mathrm{t}_S \zeta = 0, \tag{2.119}$$

$$\mathrm{t}_{S^*} \star \mathrm{d}\alpha = -\mathrm{t}_{S^*} \star \zeta. \tag{2.120}$$

In section 3.5.3 we show how this boundary value problem can be solved using the finite element method.

We now summarize the isomorphisms in a diagram [28]:

$$\begin{array}{ccccccc}
\mathcal{H}^k(M,S) & \longleftrightarrow & H^k_{\mathrm{dR}}(M,S) & \longleftrightarrow & H_k(M,S;\mathbb{F}) & \longleftrightarrow & H^k(M,S;\mathbb{F}) \\
\updownarrow \star & & & & & \swarrow\searrow & \\
\mathcal{H}^{n-k}(M,S^*) & \longleftrightarrow & H^{n-k}_{\mathrm{dR}}(M,S^*) & \longleftrightarrow & H_{n-k}(M,S^*;\mathbb{F}) & \longleftrightarrow & H^{n-k}(M,S^*;\mathbb{F})
\end{array} \tag{2.121}$$

The leftmost isomorphism is realized by the Hodge operator, and the cross on the right is the generalization of the Lefschetz duality theorem.

2.4.5 Whitney forms

As seen in the previous sections, differential forms can be regarded as cochains. In this section, we describe a map from k-cochains to k-forms that is isomorphic to its image, i.e. an injection. Such map is called the *Whitney map* and the k-forms in its image are called *Whitney forms*.

Whitney forms play a fundamental role in the finite element method where the unknown field is a differential k-form. The study of such finite element methods is nowadays called *Finite element exterior calculus*, in which the use of so-called *polynomial finite element differential forms* is studied [2, 3]. Within that framework, Whitney forms are an example of polynomial differential forms of lowest degree. Later in this thesis, Whitney forms are employed extensively. However, the general polynomial differential forms are not addressed later in this thesis, as their construction and the generalization of the *Whitney map* is not straightforward. Also, we only consider Whitney forms on a simplicial finite element mesh.

Let M be an n-manifold with a submanifold $S \subset \partial M$ and let them be covered with cell complexes. Whitney forms are defined via the finite element (E^n, ϕ_{σ^n}) shape functions λ of first polynomial order. Whitney 0-form $\mathsf{w}_0^i \in L^2\Omega^0(M)$ associated with a 0-cell $\sigma_i^0 \in M$ is

$$\mathsf{w}_0^i := \sum_{\sigma_j^n} (x^{-1} \circ \phi_{\sigma_j^n})^* \lambda_j^i, \tag{2.122}$$

where the sum is taken over n-cells σ_j^n of M that have σ_i^0 as a face. Here, we define Whitney k-forms on a regular CW-complex that is a simplicial complex.

If an m-simplex σ_i^m in M has vertices $(v_0, v_1, \dots v_m)$, its associated Whitney k-form, $k \leq m$, is

$$\mathsf{w}_k^i := m! \sum_{j=0}^{m} (-1)^m \mathsf{w}_0^j \mathrm{d}\mathsf{w}_0^{v_0} \wedge \ldots \wedge \mathrm{d}\mathsf{w}_0^{v_{j-1}} \wedge \mathrm{d}\mathsf{w}_0^{v_{j+1}} \wedge \ldots \wedge \mathrm{d}\mathsf{w}_0^{v_m}, \tag{2.123}$$

i.e. $\mathrm{d}\mathsf{w}_0^{v_j}$ is omitted in each summand [63]. For example, the most frequently used in the applications is the Whitney 1-form

$$\mathsf{w}_1^i = \mathsf{w}_0^{v_0} \mathrm{d}\mathsf{w}_0^{v_1} - \mathsf{w}_0^{v_1} \mathrm{d}\mathsf{w}_0^{v_0} \tag{2.124}$$

associated with a 1-cell σ_i^1 of M.

The vector space of **Whitney k-forms** is defined as

$$W^k(M, S) := \operatorname*{span}_{\sigma_k^i \in M \setminus S} \{\mathsf{w}_i^k\} \subset H\Omega^k(M, S), \tag{2.125}$$

i.e. the span of Whitney forms associated with k-cells in M that are not in S. Note that for $\omega \in W^k(M, S)$, $\mathrm{t}_S \omega = 0$ holds.

Whitney map $w : C^k(M, S) \to W^k(M, S) \subset H\Omega^k(M, S)$ maps a k-cochain $\gamma = \sum_i \gamma_i \epsilon^i$ to a Whitney k-form. It is defined as

$$w(\gamma) := \sum_i \gamma_i \mathsf{w}_k^i, \tag{2.126}$$

where w_k^i is a Whitney k-form associated with a k-cell σ_i^k of M. Its left-inverse $w^{-1} : L^2\Omega^k(M, S) \to C^k(M, S)$ is simply given by

$$w^{-1}(\omega) := \sum_i \Big(\int_{\sigma_i^k} \omega \Big) \epsilon^i. \tag{2.127}$$

This is indeed a left-inverse since Whitney k-forms have the property that $\int_{\sigma_i^k} \mathsf{w}_k^j = \delta_i^j$ holds with a Whitney k-form w_k^j associated with a k-cell σ_j^k. If one restricts the domain of w^{-1} to the Whitney space $W^k(M,S)$, the Whitney map becomes an isomorphism.

Another important property of Whitney spaces is that they constitute a chain complex, called **Whitney complex**, since $\mathrm{d}\omega \in W^{k+1}(M,S)$ holds for all $\omega \in W^k(M,S)$ [8]. If this property didn't hold, it would have severe consequences on the stability and the convergence of the finite element method [3]. The property can be expressed as a commutative diagram:

$$\begin{array}{ccccccc} \cdots & \xrightarrow{\delta} & C^k(M,S) & \xrightarrow{\delta} & C^{k+1}(M,S) & \xrightarrow{\delta} & \cdots \\ & & \downarrow{\scriptstyle w} & & \downarrow{\scriptstyle w} & & \\ \cdots & \xrightarrow{\mathrm{d}} & W^k(M,S) & \xrightarrow{\mathrm{d}} & W^{k+1}(M,S) & \xrightarrow{\mathrm{d}} & \cdots \end{array} \tag{2.128}$$

As an important consequence, each de Rham class $[\omega] \in H^k_{\mathrm{dR}}(M,S)$ can be represented as a an element of the Whitney space $\omega \in W^k(M,S)$. It is as important to note that the Hodge operator of differential k-forms does not map an element of the Whitney space to an element of a Whitney space. Consequently, a harmonic k-form $\omega \in \mathcal{H}^k(M,S)$ cannot accurately be represented as a Whitney k-form. However, using the finite element method, one can approximate a harmonic k-form by an element of the Whitney space $W^k(M,S)$.

Chapter 3

Homology and cohomology computation of finite element meshes

In the finite element method, the modeling domain is covered with a mesh. With each element of the mesh and with face of the elements one can associate a cell. Chains are introduced as formal sums of such cells. Chains constitute abelian groups, and they form a chain complex.

From the integer matrices that represent the boundary homomorphism of the chain complex, the homology and the cohomology of the chain complex can be computed. However, the bases of the k-chain groups have as many elements as the finite element mesh has k-faces. The computational complexity of homology and cohomology computation is cubic with respect the size of the bases. Therefore, the size of resulting chain complex should be reduced before the actual homology and cohomology computation takes place.

There are two conceptual approaches to achieve smaller problem size for the homology computation. The first is to reduce the number of cells in the cell complex before constructing the chain complex. The cell complex reduction are designed to keep the homotopy type of the underlying space invariant, and thus preserve the homology and cohomology of the chain complexes.

The second approach is to perform the reduction at the chain complex level. At each step, the chain complex is mapped to a smaller one with a chain equivalence that keeps the homology and cohomology groups of the chain complex invariant. As any homotopy equivalence induces a chain equivalence, but the converse it not true, the reduction at chain complex level is more general and used in this thesis. The problem is to find chain equivalences that are efficient to apply and reduce the chain complex as much as possible while keeping the homology and cohomology group generators of the reduced chain complex also as the generators in the original chain complex.

Once the chain complex is reduced, the actual group theoretic problems are solved. The problems are to find kernel and image groups of the boundary homomorphisms, solve for the inclusion map from the kernel to the image group, and finally find the quotient group induced by the inclusion map from the kernel to the image group. As the boundary homomorphisms are represented by integer matrices, an integer matrix decomposition called *Smith normal form* [59] is employed for the computation.

The results of the homology and cohomology computation are sets of chains that represent a generating set of the homology and cohomology groups. As the k-chains are formal sums of k-cells, i.e. k-faces of mesh elements, the results can be expressed visually in terms of the original finite element mesh.

As a second theme of this chapter, we discuss methods to control what kind of homology and cohomology generators one will obtain. Such methods are of interest in engineering, since the generators are often related to some of the key quantities of the engineering problem at hand.

3.1 Construction of chain complexes from a finite element mesh

The finite element mesh is a collection of convex polyhedral elements in $\mathbb{R}^n$ with nonempty interiors. The intersection of any two elements is required to be either empty or a common face of each of some dimension. Therefore, a finite element mesh can be interpreted as a regular CW-complex, from which a chain complex can be constructed.

3.1.1 Data structures and construction of the cell complex

Before the actual homology and cohomology computation can commence, chain complexes must be constructed from the provided finite element mesh. The data structures that represent cells and chain complexes are described next. The data structures should be such that they can be efficiently constructed and reduced.

Let M be an n-manifold that is represented by a mesh of finite elements (E^n, ϕ), i.e. $\bigcup_i \phi_i(E^n) = x(M) \subset \mathbb{R}^n$ cover a global chart (M, x) of M. The data structure that represents a finite element mesh is able to produce an indexed list of its vertices as coordinate n-tuples of $\mathbb{R}^n$, and an indexed list of its elements as integer tuples which denote the indices of the vertices of the element. There can be additional information associated to the elements that for example denote regions of the mesh where the elements belong.

Therefore, we take that a positively oriented k-cell, or a finite element, $\sigma^k = (E^k, \phi)$ is represented by an ordered list of vertex indices, where the order determines the orientation of the corresponding Euclidean reference k-cell E^k. That is, an oriented k-cell is represented by a multi-index $I(\sigma^k) = (v_1, v_2, \ldots, v_m)$, $v_i \in \mathbb{N}$. If a $k-1$-cell σ^{k-1} is on the boundary of σ^k it is represented by a permutation of a proper subset $I(\sigma^{k-1})$ of vertex indices $I(\sigma^k)$ of σ^k. The incidence relation $\iota : \mathcal{S}^k \times \mathcal{S}^{k-1} \to \{0, 1, -1\}$ of cells is then given by the Kronecker delta of multi-indices:

$$\iota(\sigma_i^k, \sigma_i^{k-1}) = \delta^{I(\sigma_i^k)}_{I(\sigma_j^{k-1})} \tag{3.1}$$

When $\iota(\sigma_i^k, \sigma_j^{k-1}) = \pm 1$, a $k-1$-cell σ_j^{k-1} is on the boundary of σ_i^k which is on the coboundary of σ_j^{k-1}. The incidence relation ι can be represented by a sparse incidence matrix D_k, whose elements are either 0, 1, or -1.

A cell complex $K = (M, \mathcal{K}, \iota)$ is represented by a list of its oriented cells in each dimension. The cells $\sigma \in \mathcal{K}$ are flagged whether they also belong to the complex $L = (S, \mathcal{L}, \iota)$ of the submanifold S.

Since the reduction algorithms in the next section require a rapid access to both the boundary and coboundary cells of a cell, two lists are associated to each cell: The list of oriented cells on its boundary and on the coboundary with the sign assigned by the incidence relation ι. That is, the the non-zero elements of both D_k and D_k^T are effectively being stored.

The algorithm 3.1.1 constructs a cell complex given a finite element mesh representation of a manifold. In the construction of the cell complex only the combinatorial information of the mesh is used. That is, the provided elements-to-vertex map, while the vertex coordinates are not needed.

Algorithm 3.1.1: Construct a cell complex from a finite element mesh

Input: Finite element mesh of a global chart image $x(M) \subset \mathbb{R}^n$ of the n-manifold M
Output: Cell complex $K = (M, \mathcal{K}, \iota)$
$\mathcal{K} := \emptyset$;
forall the *mesh elements of* M **do**
    create an oriented k-cell σ^k from a k-dimensional mesh element;
    if $\sigma^k \notin \mathcal{K}$ **then**
        $\mathcal{K} := \mathcal{K} \cup \sigma^k$;
        if $\sigma^k \in S$ **then**
            set flag indicating that $\sigma^k \in \mathcal{L}$ holds;
for $k = n$ ***to*** 1 **do**
    foreach *oriented* k***-cell*** σ^k ***in*** $\mathcal{K}$ **do**
        create all $k-1$-cells σ^{k-1} on the boundary of σ^k;
        foreach $k-1$***-cell*** σ^{k-1} ***on the boundary of*** σ^k **do**
            if $\sigma^{k-1} \notin \mathcal{K}$ ***and*** $-\sigma^{k-1} \notin \mathcal{K}$ **then**
                $\mathcal{K} := \mathcal{K} \cup \sigma^{k-1}$;
                if $\sigma^k \in S$ **then**
                    set flag indicating that $\sigma^{k-1} \in \mathcal{L}$;
            $\iota(\sigma^k, \sigma^{k-1}) := \pm 1$;

3.1.2 Construction of chain complexes

Once a cell complex K of the mesh covering the global chart image $x(M) \subset \mathbb{R}^n$ has been constructed, it is interpreted as three chain complexes $\mathcal{C}(M) = (C_\bullet(M), \partial_\bullet)$, $\mathcal{C}(S) = (C_\bullet(S), \partial_\bullet)$, and $\mathcal{C}(M, S) = (C_\bullet(M, S), \partial_\bullet)$ with the cell bases as described in the section 2.3.1 on page 38. A chain complex $\mathcal{C} = (C_\bullet, \partial_\bullet)$ is represented by the bases Σ_k of its k-chain groups C_k and by the matrix representations D_k of the boundary homomorphisms ∂_k.

The bases Σ_k of the chain groups $C_k(M)$ have the decomposition

$$\Sigma_k = (\sigma_1^k, \sigma_2^k, \ldots, \sigma_{q_k(M/S)}^k, \sigma_{q_k(M/S)+1}^k, \ldots \sigma_{q_k(M)}^k) = (\bar{\Sigma}_k, \acute{\Sigma}_k), \tag{3.2}$$

where $q_k(M/S)$ indicates the number of k-cells that are in M but not in S. In the decomposition, $\acute{\Sigma}_k$ and $\bar{\Sigma}_k$ form cell bases for chain groups $C_k(M,S)$ and $C_k(S)$, respectively.

The matrix representation D_k of the incidence relation ι of the k-cells has the block form

$$\mathrm{D}_k = \begin{bmatrix} \bar{\mathrm{D}}_k & \mathrm{A}_k \\ \mathrm{B}_k & \acute{\mathrm{D}}_k \end{bmatrix}, \tag{3.3}$$

which also serves as the representation of the boundary homomorphism $\partial_k : C_k(M) \to C_{k-1}(M)$. The boundary homomorphisms of chain complexes $(C_\bullet(M,S), \partial_\bullet)$ and $(C_\bullet(S), \partial_\bullet)$ are then represented by the matrices $\bar{\mathrm{D}}_k$ and $\acute{\mathrm{D}}_k$, respectively, see section 2.3.1 (page 38).

In summary, a representation of a cell complex K where the cells that belong to the cell subcomplex L are flagged can be interpreted as any of the three chain complexes with no additional computational cost. Since the representation of the coboundary homomorphism δ_k is D_{k+1}^T, also the cochain complexes can be generated from the cell complex representation.

Remark 3.1.1. All the chain groups in the chain complexes are finitely generated free abelian by the construction of the section 2.3.1.

3.2 Reduction of chain complexes

The reduction of a chain complex relies on chain equivalences. A chain equivalence transforms the chain groups of a chain complex to ones with smaller bases in such manner that the homology and the cohomology groups of the chain complex are preserved, as described in section 2.1.2 on page 22.

There are multiple chain equivalences with different characteristics, but here we concentrate on a class of them that removes a *reduction pair* from the chain complex. Such chain equivalences are attractive since reduction pairs can be quickly detected by looking at the local boundary information of the basis cells of the chain groups. Reduction pairs first appeared in [39], whose definitions and propositions are cited in the following subsection.

The idea of the chain complex reduction is the successive removal of reduction pairs from the chain complex. At each recursion step, the total number of basis chains in the reduced chain complex is two less than its predecessor's.

Although in this thesis the bases of the chain groups are cell bases of the cells belonging to a manifold, the reduction methods described in this section apply to any abstract chain complex consisting of finitely generated free abelian groups.

3.2.1 Reduction algorithms

There are three requirements for the algorithms and data structures that rely on the reduction pairs:

1. The algorithm should be able to quickly detect and remove a reduction pair from the chain complex.

2. The smaller the number of the basis chains in the final chain complex the better.

3. The representative chains of the generators of the homology or the cohomology groups of the reduced chain complex should also represent the generators of the homology or cohomology groups of the original chain complex.

Formally, the third requirement is stated as follows. Let $i_k : \check{C}_k \to C_k$ be an injective map from the reduced chain groups to the chain groups of the original chain complex. Then, one requires that if $[\check{z}] \in H_k(\check{\mathcal{C}})$ holds, then $[i_k \check{z}] \in H_k(\mathcal{C})$ must hold. On the cohomology side, if $[\check{\zeta}] \in H^k(\check{\mathcal{C}}; G)$ holds, then $[\tilde{i}_k^{-1} \check{\zeta}] \in H^k(\mathcal{C}; \mathbb{Z})$ must hold.

The third requirement differentiates the reduction algorithms to homology reduction algorithms and cohomology reduction algorithms. The former retain the homology group representatives and the latter retain the cohomology group representatives. An algorithm that would retain both representatives would necessitate a trade-off with the second requirement. That is, the number of basis chain in the reduced chain groups would be larger. This results from the fact that the support of homology and cohomology generators together is larger than the support of the other generators alone. See Fig. 3.1 for an example.

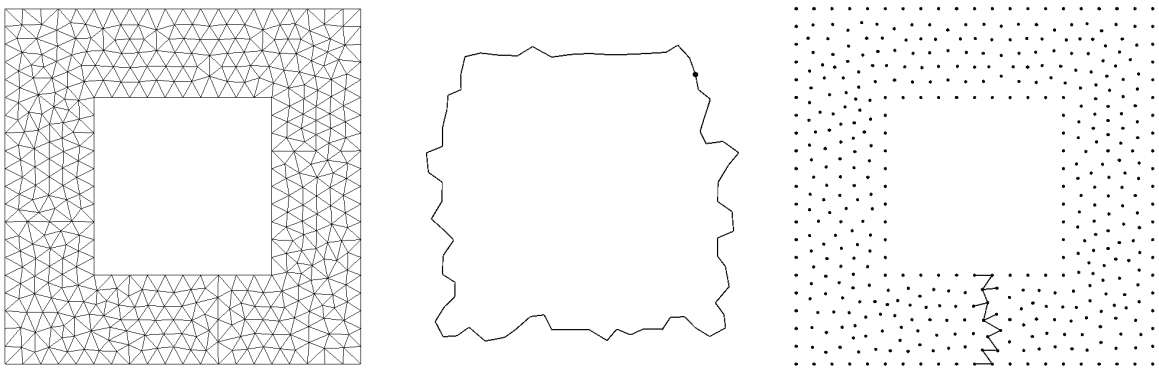


Figure 3.1: On the right: the original finite element mesh. On the center: Chain complex after homology reduction. On the left: Chain complex after cohomology reduction. Both homology and cohomology reduced chain complexes have one basis 0-chain and one basis 1-chain.

3.2.2 Reduction pair

Let Σ_k for each k denote bases for the chain groups of the chain complex $\mathcal{C} = (C_\bullet, \partial_\bullet)$. A **$q$-reduction pair** (b, a) is a pair of basis chains such that $a \in \Sigma_{q-1}$ is on the boundary of $b \in \Sigma_q$. That is, $\langle \partial b, a \rangle = \pm 1$ holds, see section 2.3.1.

The **reduced chain complex** $\check{\mathcal{C}} = (\check{C}_\bullet, \check{\partial}_\bullet)$ of a chain complex $\mathcal{C}$ where the reduction pair has been removed is such that

$$\check{C}_k := \begin{cases} C_k & \text{if } k \neq q \text{ and } k \neq q-1 \\ \{c \in C_k \mid \langle c, b \rangle = 0\} & \text{if } k = q \\ \{c \in C_k \mid \langle c, a \rangle = 0\} & \text{if } k = q-1 \end{cases} \tag{3.4}$$

and

$$\check{\partial}_k c := \begin{cases} \partial c & \text{if } k \neq q \text{ and } k \neq q+1 \\ \partial c - \frac{\langle \partial c, a \rangle}{\langle \partial b, a \rangle} \partial b & \text{if } k = q \\ \partial c - \langle \partial c, b \rangle b & \text{if } k = q+1 \end{cases} \qquad \forall\, c \in \check{C}_k \tag{3.5}$$

hold. The bases $\check{\Sigma}_k$ of $\check{C}_k$ are

$$\check{\Sigma}_k := \begin{cases} \Sigma_k & \text{if } k \neq q \text{ and } k \neq q-1 \\ \Sigma_k \setminus \{b\} & \text{if } k = q \\ \Sigma_k \setminus \{a\} & \text{if } k = q-1 \end{cases} . \tag{3.6}$$

That is, the basis elements a and b removed from the bases Σ_q and Σ_{q-1}, and the boundary operators are adjusted accordingly. It is proved in [39] that $\check{\mathcal{C}}$ qualifies as a chain complex.

Remark 3.2.1. The boundary of a basis k-chain in the reduced chain complex doesn't necessarily correspond to the topological boundary of its underlying k-cell. That is, one cannot equate a basis k-chain with a basis k-cell after reduction. However, they remain related.

Define a chain map $p_k : C_k \to \check{C}_k$ by

$$p_k c := \begin{cases} c & \text{if } k \neq q \\ c - \frac{\langle c, a \rangle}{\langle \partial b, a \rangle} \partial b & \text{if } k = q-1 \\ c - \langle c, b \rangle b & \text{if } k = q \end{cases} , \quad c \in C_k. \tag{3.7}$$

It turns out that p_k is a chain equivalence if one defines a chain map $j_k : \check{C}_k \to C_k$ by

$$j_k c := \begin{cases} c & \text{if } k \neq q \\ c - \frac{\langle \partial c, a \rangle}{\langle \partial b, a \rangle} b & \text{if } k = q \end{cases} , \quad c \in \check{C}_k. \tag{3.8}$$

Then, the map $p_k \circ j_k : \check{C}_k \to \check{C}_k$ is the identity map of $\check{C}_k$ for each k, and the map $j_k \circ p_k : C_k \to C_k$ is chain homotopic to the identity map of C_k, i.e. the map p_k is a chain equivalence. That is, there exist chain maps $D_k : C_k \to \check{C}_{k+1}$ for which

$$\check{\partial}_{k+1} \circ D_k + D_{k-1} \circ \partial_k = j_k \circ p_k - \mathrm{id} \tag{3.9}$$

holds. The maps D_k are defined by

$$D_k c := \begin{cases} c - \frac{\langle c, a \rangle}{\langle \partial b, a \rangle} b & \text{if } k = q-1 \\ 0 & \text{otherwise} \end{cases} , \quad c \in C_k. \tag{3.10}$$

The proofs for the above statements are direct substitutions and can be found in [39].

Since the map p_k is a chain equivalence, the induced map $p_* : H_k(\mathcal{C}) \to \check{H}_k(\mathcal{C})$ is an isomorphism from the k:th homology group $H_k(\mathcal{C})$ of $\mathcal{C}$ to the k:th homology group $H_k(\check{\mathcal{C}})$ of $\check{\mathcal{C}}$. The inverse of p_* is the induced map $j_* : H_k(\check{\mathcal{C}}) \to H_k(\mathcal{C})$ of j_k. Furthermore, the induced map $p^* : H^k(\mathcal{C}; \mathbb{Z}) \to H^k(\check{\mathcal{C}}; \mathbb{Z})$ is also an isomorphism. That is, ***the homology and the cohomology groups of the chain complexes $\mathcal{C}$ and $\check{\mathcal{C}}$ are isomorphic***.

Remark 3.2.2. The homology and the cohomology groups of the original chain complex and the reduced chain complex are *merely* isomorphic. That is, it is not guaranteed that the representative chain of the element in the homology group $H_k(\check{\mathcal{C}})$ of the reduced chain complex represents an element of the homology group $H_k(\mathcal{C})$ of the original chain complex.

Elementary reduction procedure

All the reduction algorithms use the `removeReductionPair` routine which removes a reduction pair (b, a) from a chain complex $\mathcal{C}$. Its computational complexity $R(|\Sigma_k|, |\Sigma_{k-1}|)$ depends on the data structure that is used to store the basis Σ_k of k-chains of the chain group C_k. For example, if the data structure is a list, a *binary search tree*, or a *hash map*, the complexity $R(|\Sigma_k|, |\Sigma_{k-1}|)$ to remove a reduction pair is $O\Big(\max(|\Sigma_k|, |\Sigma_{k-1}|)\Big)$, $O\Big(\log(\max(|\Sigma_k|, |\Sigma_{k-1}|))\Big)$ or $O\Big(1\Big)$, respectively.

Procedure removeReductionPair

Input: A chain complex $\mathcal{C}$ and a k-reduction pair (b, a)
Output: A reduced chain complex $\check{\mathcal{C}}$
$\check{\mathcal{C}} := \mathcal{C}$;
$\check{\Sigma}_k := \check{\Sigma}_k \setminus \{b\}$;
$\check{\Sigma}_{k-1} := \check{\Sigma}_{k-1} \setminus \{a\}$;
forall the k***-chains*** $c_k \in \check{\Sigma}^k$ **do**
  $\check{\partial}_k c := \partial c - \frac{\langle \partial c, a \rangle}{\langle \partial b, a \rangle} \partial b$;
forall the $k+1$***-chains*** $c_{k+1} \in \check{\Sigma}^{k+1}$ **do**
  $\check{\partial}_{k+1} c := \partial c - \langle \partial c, b \rangle b$;

Table 3.1: Reduction algorithms.

Name	Complexity	Homology/ Cohomology	Homotopy invariant
k-collapse	$O\Big(q_{k-1} \cdot R(q_k, q_{k-1})\Big)$	Homology	Yes
k-combine	$O(q_k q_{k-1})$	Homology	Yes
Omit-collapse	$O\Big(q_{k-1} \cdot R(q_k, q_{k-1})\Big)$	Homology	No
k-dual-collapse	$O\Big(q_{k-1} \cdot R(q_k, q_{k-1})\Big)$	Cohomology	Dual
k-dual-combine	$O(q_k q_{k-1})$	Cohomology	Dual
Coreduction	$O\Big(q_{k-1} \cdot R(q_k, q_{k-1})\Big)$	Cohomology	No

3.2.3 Homology reduction algorithms

For the homology reduction algorithms it holds true that if $[\check{z}]$ is a generator of $H_k(\check{\mathcal{C}})$, then $[i_k \check{z}]$ is a generator $H_k(\mathcal{C})$, where $i_k : \check{C}_k \to C_k$ is an inclusion map.

k-collapse

We now introduce a homology reduction algorithm called **k-collapse**. It appears in [39, 61, 49]. To interpret it geometrically, it "pushes-in" a reduction pair, see Fig. 3.2. If applied to a chain complex $\mathcal{C}(M)$ of a manifold M, the algorithm keeps the homotopy type of the space underlying the basis chains invariant. Therefore, the chains that represent the generators of the homology groups of the reduced chain complex also represent the generators of the original chain complex.

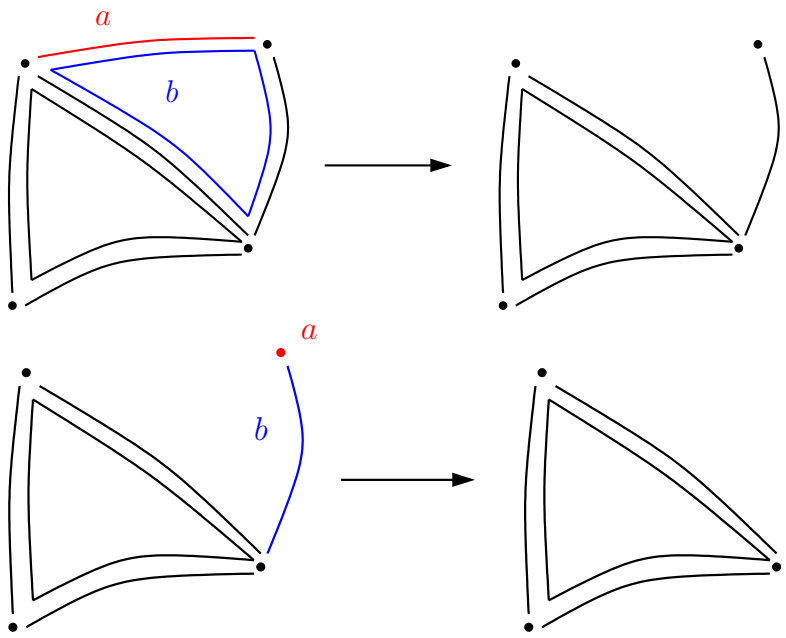


Figure 3.2: 2-collapse and 1-collapse of a chain complex.

Algorithm 3.2.1: k-collapse

Input: A chain complex $\mathcal{C}$
Output: A reduced chain complex $\mathcal{C}$
repeat
  forall the *basis* $k-1$-*chains* $a \in \Sigma_{k-1}$ **do**
    if a ***is on the boundary of exactly one basis*** k-***chain*** $b \in \Sigma_k$ **then**
      $\mathcal{C} :=$ removeReductionPair$(\mathcal{C}, (b, a))$;
until *no reduction pair* (b, a) ***has been removed from*** $\mathcal{C}$;

The worst case computational complexity of the k-collapse algorithm is quadratic. However, one quickly notices that for a chain complex that is constructed from a typical finite element mesh, the repeat-until-loop is only ran a couple of times. This is affected by the order in which the reduction pairs are removed from the chain complex. The order is determined how the cell and chain complexes were constructed from the finite element mesh and how the mesh vertices were indexed by the mesh generator. That is,

one might study the effects of mesh reindexing heuristics to the average performance of the k-collapse algorithm.

If the repeat-until-loop is ran only a couple of times, the computational complexity of the algorithm is determined by the complexity $R(|\Sigma_k|, |\Sigma_{k-1}|)$ of the `removeReductionPair` routine. Then, the computational complexity of the k-collapse algorithm is $O\Big(|\Sigma_{k-1}| \cdot R(|\Sigma_k|)\Big)$ since the for-loop is ran at least once.

Let the k-collapse algorithm be applied to a relative chain complex $\mathcal{C}(M, S)$ of a manifold M with submanifold $S \subset \partial M$ with cell bases. Without a loss of generality, we assume that M has only one connected component. If it had many, the following analysis will apply to them individually.

The relation of the boundary ∂M to the submanifold S has a decisive role how many reduction pairs (b, a) the algorithm is able to remove from the chain complex $\mathcal{C}(M)$.

The first reduction pair the k-collapse algorithm can find always includes a basis $k-1$-chain that corresponds to a $k-1$-cell on the boundary. Usually, just one reduction pair removal triggers an avalanche of other removals. In two cases, k-collapse cannot remove a single reduction pair from a relative chain complex $\mathcal{C}(M, S)$:

1. The manifold M is closed and the submanifold S is empty. That is, $\partial M = S = \emptyset$ hold. For if a basis $k-1$-chain a would be on the boundary of a single basis k-chain b, it would correspond a $k-1$-cell that lies on the boundary ∂M of the manifold M, which is empty.

2. The submanifold S equals the boundary ∂M of the manifold M. That is, $\partial M = S$ holds. Similarly to the first case, a candidate basis $k-1$-chain a would correspond to a $k-1$-cell on the boundary ∂M, but all such basis chains belong to the chain group $C_{k-1}(S)$, and are thus absent from the basis of $C_{k-1}(M, S)$.

That is, while being efficient, the k-collapse algorithm cannot start in these two important special cases.

k-combine

In the **k-combine** algorithm one uses a different rule to find reduction pairs. It also appears in [39, 61, 49]. When the algorithm is interpreted geometrically, the removal of a reduction pair from the chain complex corresponds to "combining" two k-cells of the cell complex and a removal of their common face $k-1$-cell, see Fig. 3.3. In the geometric interpretation, the homotopy type of the underlying space is kept invariant.

However, when interpreted on the chain complex level, the other k-cell is just removed from the basis of the k-chain group, but the boundary of the remaining basis k-chain adjusted according to (3.5).

In order to be able to construct the chains that represent the generators of the homology groups in the original chain complex from the chains that represent the generators in the reduced chain complex, some extra memory is sacrificed in the implementation. The basis k-chain will remember from which k-cells it was combined from.

The k-combine algorithm has worst-case computational complexity of $O\Big(|\Sigma_k| \cdot |\Sigma_{k-1}|\Big)$ which typically realizes since eventually, most of the basis $k-1$-chains get enqueued in Q. The algorithm could be implemented without the queue, but based on test runs,

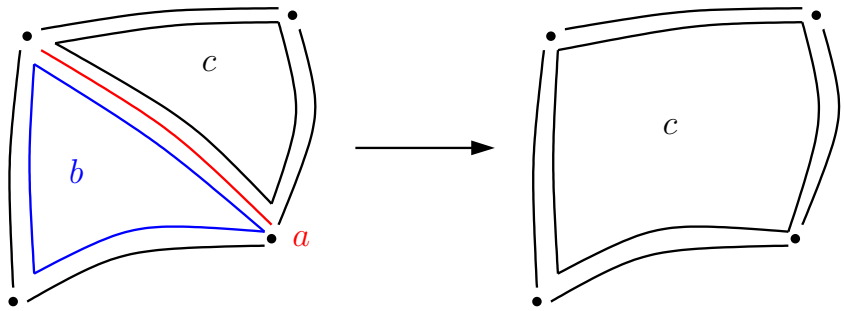


Figure 3.3: 2-combine of a chain complex.

Algorithm 3.2.2: k-combine

Input: A chain complex $\mathcal{C}$
Output: A reduced chain complex $\mathcal{C}$
$Q :=$ empty queue of generator $k-1$-chains;
forall the *basis* k-*chains* $c \in \Sigma_k$ **do**
  enqueue all basis $k-1$-chains on the boundary of c to Q;
  while Q ***is not empty*** **do**
    dequeue a from Q;
    if a ***is on the boundary of exactly two basis*** k-***chains*** b ***and*** c **then**
      enqueue generator $k-1$-chains $c \neq a$ on the boundary of b and c not already in Q to Q;
      $\mathcal{C} :=$ removeReductionPair($\mathcal{C}$, (b, a));

such version is not able to remove as many reduction pairs from a chain complex that is constructed from a typical finite element mesh. Heuristically, in the queued version the algorithm grows a "mother" basis k-chain that absorbs k-cells around it. In the version without a queue, the algorithm grows a lot of medium-sized basis k-chains whose combinations no longer happen. As with the k-collapse algorithm, ultimately the number of reduction pairs the k-combine algorithm is able to remove from the chain complex depends on the indexing of the mesh vertices.

Omit-collapse

The observation that the k-collapse algorithm is toothless in two cases leads to an algorithm called **Omit-collapse** in which the k-collapse algorithm is a subroutine. The Omit-collapse algorithm was originally proposed in [49], and it can be considered to be the dual algorithm to the coreduction algorithm proposed in [47].

The two cases where the k-collapse algorithm cannot proceed have a similarity. Let M be an ***orientable*** n-manifold with a single connected component and $S \subset \partial M$ its submanifold. If $\partial M = S$ holds, then $\dim H_n(M, S) = 1$ holds by the Lefschetz and de Rham duality theorems: $H_n(M, \partial M) \simeq H_0(M)$.

Consider the two cases where the k-collapse algorithm fails to remove any reduction pairs. Let M be an ***orientable*** n-manifold with a single connected component and

$S \subset \partial M$ its submanifold. The k-collapse algorithm failed when $\partial M = S$ or $\partial M = \emptyset$ held.

Observe that $\dim H_n(M, \partial M) = 1$ holds, i.e $H_n(M, S) \simeq \mathbb{Z}$ holds by the duality $H_n(M, \partial M) \simeq H_0(M)$. Moreover, the single generator of $H_n(M, \partial M)$ is represented by a multiple of the sum of all the basis n-cells in the group $C_n(M, S)$ [31]. Then, one can omit the computation of the generator of the n:th homology group $H_n(M, \partial M)$ since it is already known. This observation is dual to the fact that one can produce a generator representation for the 0:th homology group $H_0(M)$ of any manifold M just by picking a 0-cell from each connected component of M.

The Omit-collapse algorithm exploits the fact that the failure of the k-collapse algorithm yields this information about the group $H_n(M, S)$. It first executes k-collapse for all k. If there are basis n-chains left in $C_n(M, S)$ after n-collapse, $\dim H_n(M, S) \neq 0$ is know to hold.

Then, a basis $n+1$-chain b is added to the previously empty chain group $C_{n+1}(M, S)$ and define ∂_{n+1} to be such that $\partial_{n+1} b = a$ holds for a basis n-chains a in $C_n(M, S)$.

We can then remove the reduction pair (b, a) from the chain complex, and n-collapse algorithm can remove more basis n-chains from the chain complex. The basis n-chains removed by the n-collapse together with the n-chain a constitute a representative n-chain z^n of the generator of the group $H_n(M, S)$.

If M has more than one connected component, this procedure is repeated until there are no basis n-chains left in the chain complex. Finally, we consider the n-chains z_i^n that represent elements of $H_n(M, S)$ as a basis for the group $C_n(M, S)$ and set the boundary homomorphism to $\partial_n = 0$.

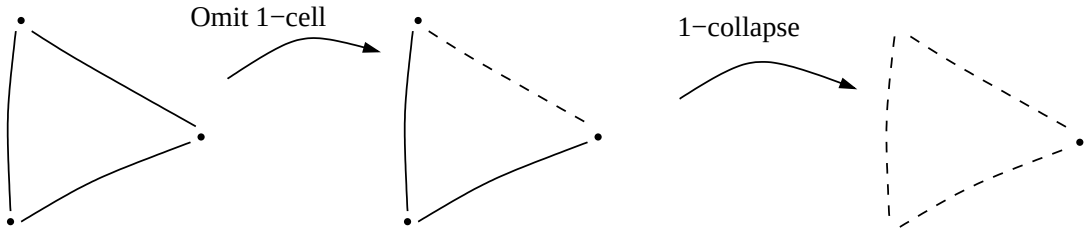


Figure 3.4: Omit-collapse of a chain complex constructed from a closed 1-manifold.

3.2.4 Cohomology reduction algorithms

For the cohomology reduction algorithms it holds true that if $[\check{\zeta}]$ is a generator of $H^k(\check{\mathcal{C}}; \mathbb{Z})$, then $[\tilde{i}_k^{-1}\check{\zeta}]$ is a generator $H^k(\mathcal{C}; \mathbb{Z})$, where $i_k : \check{C}_k \to C_k$ is an inclusion map.

The k-collapse and k-combine algorithms have "dual" cohomology reduction algorithms in the sense that they rely on the coboundary information of the basis chains instead of the boundary information. As k-collapse and k-combine could be interpret to keep the homotopy type of the underlying space invariant, their dual algorithms keep the space underlying the *dual cell complex* [48] homotopy invariant.

The dual k-collapse algorithm has a similar problem as k-collapse algorithm. Let M be an n-manifold with a single connected component and let $S \subset \partial M$ be its submanifold.

Algorithm 3.2.3: Omit-collapse

Input: A chain complex $\mathcal{C}$
Output: A reduced chain complex $\mathcal{C}$
foreach $k = n$ ***to 1*** **do**
  run k-collapse;
$i := 1$;
while $|\Sigma_n| \neq 0$ **do**
  pick a basis n-chain $a \in \Sigma^n$;
  $\Sigma_{n+1} := \{b\}$;
  $\partial_{n+1} b := a$;
  $\mathcal{C} :=$ removeReductionPair($\mathcal{C}$, (b, a));
  foreach $k = n$ ***to 1*** **do**
    run k-collapse;
  create an n-chain z_i that is the sum of all basis n-chains removed in the previous n-collapse;
  $z_i := a + z_i$;
  $i := i + 1$;
$\Sigma_n := \bigcup_i \{z_i\}$;
$\partial_n := 0$;

Then, if $S = \emptyset$ holds, the dual k-combine algorithm cannot remove any basis cells from the chain complex $\mathcal{C}(M, S)$. Therefore, the cohomological counterpart of the Omit-collapse called *Coreduction* is represented.

Coreduction

The **Coreduction** algorithm appeared in [47] as a method for Betti number computation. In this thesis, it used as a cohomology reduction algorithm before the actual computation of the cohomology generators of a given chain complex.

If M is an n-manifold with a single connected component with a submanifold S. Then, the Coreduction algorithm cannot find any reduction pairs from a chain complex $\mathcal{C}(M, S)$ unless S is non-empty. Thus similarly to the Omit-collapse algorithm, a -1-chain a is added to the chain complex with a single basis 0-chain b on its coboundary. Then, the reduction pair (b, a) is removed from the chain complex. This triggers a number of other reduction pair removals from the chain complex, and all the cobasis 0-cochains of the removed basis 0-chains together with the cobasis element of b constitute an representative of the single generator of the cohomology group $H^0(M)$.

3.3 Computation of homology and cohomology

Let $\mathcal{C}$ be a chain complex with bases Σ_k of C_k for each k. The boundary homomorphisms ∂_k are represented by the integer matrices D_k with respect to those bases.

The goal of the homology computation is to represent the generators of the homology groups $H_k(\mathcal{C})$ in terms of the bases Σ_k. The numerical result is an integer matrix H_k,

Algorithm 3.2.4: Coreduction

Input: A chain complex $\mathcal{C}$
Output: A reduced chain complex $\mathcal{C}$
foreach $k = 0$ ***to*** n **do**
  forall the *basis* k-*chains* $c \in \Sigma_k$ **do**
    enqueue c_k to Q;
    while Q ***is not empty*** **do**
      dequeue b from Q;
      if b ***is on the coboundary of exactly one generator chain*** a **then**
        enqueue all generator chains on the coboundary of a not already in Q to Q;
        $\mathcal{C}$:= removeReductionPair($\mathcal{C}$, (b, a));
      else if ***the boundary of*** b ***is empty*** **then**
        enqueue all basis chains on the coboundary of b not already in Q to Q;

vectors $\mathbf{T}_i$, and coefficients t_i such that:

- k-chains $\Sigma_k \mathrm{H}_k = \{z_1, z_2, \ldots, z_r\}$ represent the generators of the free subgroup of $H_k(\mathcal{C})$
- k-chains $\Sigma_k \mathbf{T}_{k,i} = z_i$ represent the generator of the i:th torsion subgroup of $H_k(\mathcal{C})$ and $t_{k,i}$ is the corresponding torsion coefficient of z_i.

From the chain complex $\mathcal{C}$, one can construct a cochain complex with bases Σ^k of the groups $C^k(\mathcal{C}; \mathbb{Z})$ as described in the section 2.3.1 on page 38. By construction, the bases are such that $\Sigma^k_i(\Sigma^j_k) = \delta^i_j$ holds, i.e. Σ^k is the cobasis of Σ_k. Then, the coboundary homomorphisms δ_k are represented by the matrices D^T_{k+1} with respect to those bases.

Similarly to the homology computation, the goal of the cohomology computation is to represent the generators of the cohomology groups $H^k(\mathcal{C}; \mathbb{Z})$ in terms of the bases Σ^k. The numerical result is an integer matrix H^k, vectors $\mathbf{T}^i$, coefficients t^i such that

- k-cochains $\Sigma^k \mathrm{H}^k = \{\zeta^1, \zeta^2, \ldots, z^r\}$ represent the generators of the free subgroup of $H^k(\mathcal{C}; \mathbb{Z})$
- k-cochains $\Sigma^k \mathbf{T}^{k,i} = \zeta^i$ represent the generator of the i:th torsion subgroup of $H^k(\mathcal{C}; \mathbb{Z})$ and $t^{k,i}$ is the corresponding torsion coefficient of ζ^i.

The homology and cohomology computations involve two group theoretic problems [61]:

- Kernel-image problem: given two finitely generated free abelian groups G_1 and G_2 with bases S_1 and S_2, and a the matrix representation F of the group homomorphisms $\phi : G_2 \to G_1$, find bases for the subgroups $\ker \phi$ and $\mathrm{im}\, \phi$.

- Quotient problem: given bases S and R for two finitely generated free abelian groups G and H, respectively, find the inclusion map $i : H \to G$ and the generating set of the quotient group $G/i(H)$.

Since the diagrams

$$\begin{array}{ccc} G_2 & \xrightarrow{\phi} & G_1 \\ \updownarrow & & \updownarrow \\ \mathbb{Z}^{|S_2|} & \xrightarrow{\mathrm{F}} & \mathbb{Z}^{|S_1|} \end{array} \quad \text{and} \quad \begin{array}{ccccc} H & \xrightarrow{i} & G & \xrightarrow{j} & G/H \\ \updownarrow & & \updownarrow & & \updownarrow \\ \mathbb{Z}^{|R|} & \xrightarrow{\mathrm{I}} & \mathbb{Z}^{|S|} & \xrightarrow{\mathrm{J}} & \mathbb{Z}^r \oplus \mathbb{Z}_{t_1} \oplus \ldots \oplus \mathbb{Z}_{t_k} \end{array} \tag{3.11}$$

commute, the kernel-image and the quotient problems can be solved by starting from the matrices F and I. To solve such problems an integer matrix decomposition called *Smith normal form* is employed repeatedly.

3.3.1 Smith normal form

An integer matrix decomposition **Smith normal form** of an $m \times n$ matrix A is

$$\mathrm{A} = \mathrm{USV}, \tag{3.12}$$

where the $m \times n$ matrix U and the $n \times m$ matrix V are both **unimodular**: Their determinants are ± 1 and their inverses are integer matrices. The $m \times n$ matrix S is block diagonal:

$$\mathrm{S} = \begin{bmatrix} \mathrm{S_d} & 0 \\ 0 & 0 \end{bmatrix}, \tag{3.13}$$

where the integer $r \times r$ matrix $\mathrm{S_d}$, $r = \operatorname{rank} \mathrm{A}$, is diagonal with positive elements and its i:th entry s_i divides s_{i+1} for each $i \in \{1, 2, \ldots, r)\}$ [64]. If the group homomorphism $\mathrm{A} : \mathbb{Z}^m \to \mathbb{Z}^n$ is injective, $\mathrm{S} = \begin{bmatrix} \mathrm{S_d} & 0 \end{bmatrix}^T$ holds, if surjective, $\mathrm{S} = \begin{bmatrix} \mathrm{S_d} & 0 \end{bmatrix}$ holds.

3.3.2 Kernel-image problem

Let $\Sigma_k = (\sigma_1^k, \sigma_2^k, \ldots, \sigma_{n_k}^k)$ and $\Sigma_{k-1} = (\sigma_1^{k-1}, \sigma_2^{k-1}, \ldots, \sigma_{n_{k-1}}^{k-1})$ be bases for C_k and C_{k-1} respectively. One is to find bases for the groups $Z_k = \ker \partial_k$ and $B_{k-1} = \operatorname{im} \partial_k$. Now

$$\partial_k\big(\Sigma_k \mathbf{c}\big) = \Sigma_{k-1} \mathrm{D}_k \mathbf{c} \tag{3.14}$$

holds for all $\mathbf{c} \in \mathbb{Z}^{n_k}$, see section 2.3.1 on page 38. Then, by decomposition $\mathrm{D}_k = \mathrm{USV}$ and notation $\mathrm{V}^{-1} = \mathrm{W}$ one obtains the expression

$$\partial_k\Big(\Sigma_k \begin{bmatrix} \mathrm{W}_1 & \mathrm{W}_2 \end{bmatrix}\Big) = \Sigma_{k-1} \begin{bmatrix} \mathrm{U}_1 & \mathrm{U}_2 \end{bmatrix} \begin{bmatrix} \mathrm{S_d} & 0 \\ 0 & 0 \end{bmatrix}. \tag{3.15}$$

That is, elements $\Sigma_k \mathrm{W}_2$ map to zero under ∂_k, and elements $\Sigma_k \mathrm{W}_1$ map to integer combinations of $\Sigma_{k-1}\mathrm{U}_1\mathrm{S}_d$ under ∂_k. Therefore, $\Sigma_k \mathrm{Z}_k := \Sigma_k \mathrm{W}_2$ is a basis for the group Z_k and $\Sigma_{k-1}\mathrm{B}_{k-1} := \Sigma_{k-1}\mathrm{U}_1\mathrm{S}_d$ is a basis for the group B_{k-1}.

Similarly, let $\Sigma^k = (\epsilon_k^1, \epsilon_k^2, \ldots, \epsilon_k^{n_k})$ and $\Sigma^{k+1} = (\epsilon_{k-1}^1, \epsilon_{k-1}^2, \ldots, \epsilon_{k-1}^{n_{k-1}})$ be bases for $C^k(\mathcal{C};\mathbb{Z})$ and $C^{k+1}(\mathcal{C};\mathbb{Z})$ respectively. One is to find bases for the groups $Z^k(\mathcal{C};\mathbb{Z}) = \ker \delta_k$ and $B^{k+1}(\mathcal{C};\mathbb{Z}) = \operatorname{im} \delta_k$. Now

$$\delta_k \gamma = \delta_k\Big(\Sigma^k \boldsymbol{\gamma}\Big) = \Sigma^{k+1}\mathrm{D}_{k+1}^T \boldsymbol{\gamma} \tag{3.16}$$

holds for all $\gamma \in C^k(\mathcal{C};\mathbb{Z})$, see section 2.3.1. The problem is equivalent to the previous one with the distinction that the matrix D_k is replaced by the matrix D_{k+1}^T. Thus, in terms of the decomposition $\mathrm{D}_{k+1}^T = \mathrm{USV}$, elements $\Sigma^k \mathrm{Z}^k := \Sigma^k \mathrm{W}_2$ is a basis for the group $Z^k(\mathcal{C};\mathbb{Z})$ and elements $\Sigma^{k+1}\mathrm{B}^{k+1} := \Sigma^{k+1}\mathrm{U}_1\mathrm{S}_\mathrm{d}$ is a basis for the group $B^{k+1}(\mathcal{C};\mathbb{Z})$.

To summarize, exactly the same procedure that can be used to compute bases for Z_k and B_k can be used to compute bases for $Z^k(\mathcal{C};\mathbb{Z})$ and $B^k(\mathcal{C};\mathbb{Z})$. If the bases of C_k and $C^k(\mathcal{C};\mathbb{Z})$ agree in the sense that $\Sigma^k \Sigma_k^T = \delta_j^i$ holds for each k, the only difference is that on the cochain side, the input for the algorithm is the matrix D_{k+1}^T rather than the matrix D_k. The algorithm KerIm computes representation matrices for the bases of the groups $\ker \phi$ and $\operatorname{im} \phi$ when matrix representation of the group homomorphism ϕ is given as an input.

Procedure KerIm

Input: A matrix D

Output: Matrices Z and B

Decompose $\mathrm{D} = \begin{bmatrix}\mathrm{U}_1 & \mathrm{U}_2\end{bmatrix} \begin{bmatrix}\mathrm{S}_\mathrm{d} & 0 \\ 0 & 0\end{bmatrix} \mathrm{V}$;

Let $\begin{bmatrix}\mathrm{W}_1 & \mathrm{W}_2\end{bmatrix} := \mathrm{V}^{-1}$;

Let $\mathrm{Z} = \mathrm{W}_2$;

Let $\mathrm{B} = \mathrm{U}_1\mathrm{S}_\mathrm{d}$

3.3.3 Quotient problem

Let $\Sigma_k \mathrm{Z}_k$ be a basis for the group Z_k and $\Sigma_k \mathrm{B}_k$ be a basis for the group B_k. To solve the quotient problem, examine the short exact sequence

$$0 \to B_k \xrightarrow{i_k} Z_k \xrightarrow{j_k} H_k(\mathcal{C}) \to 0. \tag{3.17}$$

Since the group $H_k(\mathcal{C})$ is isomorphic to the group $Z_k / \operatorname{im} i_k$, see section 2.1.1 on page 18, one can deduce a representation for the generators of $H_k(\mathcal{C})$ from the representation of the image basis of B_k under i_k in a basis of Z_k. One is first to find a representation matrix for the injective map $i_k : B_k \to Z_k$ in the short exact sequence. We are to find a matrixI_k such that the equation

$$i_k\Big(\Sigma_k \mathrm{B}_k \mathbf{b}\Big) = \Sigma_k \mathrm{Z}_k \mathrm{I}_k \mathbf{b} \tag{3.18}$$

holds for all $\mathbf{b} \in \mathbb{Z}^{n_k}$. Then, by the decomposition $\mathrm{Z}_k = \mathrm{USV}$ and notation $\mathrm{U}^{-1} = \mathrm{T}$ one concludes the relation

$$\begin{bmatrix} \mathrm{T}_1 \\ \mathrm{T}_2 \end{bmatrix} \mathrm{B}_k = \begin{bmatrix} \mathrm{S_d} \\ 0 \end{bmatrix} \mathrm{VI}_k \implies \mathrm{I}_k = \mathrm{V}^{-1}{\mathrm{S_d}}^{-1}\mathrm{T}_1\mathrm{B}_k. \tag{3.19}$$

To find a generating set for the quotient group Z_k/B_k, decompose $\mathrm{I}_k = \mathrm{USV}$. The matrix S has a block form

$$\mathrm{S} = \begin{bmatrix} \mathrm{I} & 0 \\ 0 & \mathrm{D} \\ 0 & 0 \end{bmatrix} \tag{3.20}$$

where I is an identity matrix and D is a diagonal matrix with $d_i > 1$. Denote $\mathrm{V}^{-1} = \mathrm{W}$ to obtain

$$i_k\Big(\Sigma_k \mathrm{B}_k \begin{bmatrix} \mathrm{W}_1 & \mathrm{W}_2 \end{bmatrix}\Big) = \Sigma_k \mathrm{Z}_k \begin{bmatrix} \mathrm{U}_1 & \mathrm{U}_2 & \mathrm{U}_3 \end{bmatrix} \begin{bmatrix} \mathrm{I} & 0 \\ 0 & \mathrm{D} \\ 0 & 0 \end{bmatrix}. \tag{3.21}$$

Now, a basis for the free subgroup of Z_k/B_k is generated by the elements

$$j_k(\Sigma_k \mathrm{Z}_k \mathrm{U}_3) = \{[\Sigma_k \mathrm{Z}_k \mathrm{U}_3 \mathbf{z}] \in H_k(\mathcal{C}) \,|\, \mathbf{z} \in \mathbb{Z}^r\} \tag{3.22}$$

since they do not belong to $\operatorname{im} i_k$. The basis of the free subgroup of Z_k/B_k has thus a matrix representation $\mathrm{H}_k = \mathrm{Z}_k\mathrm{U}_3$.

The torsion subgroup of Z_k/B_k is generated by the elements

$$j_k(\Sigma_k \mathrm{Z}_k \mathrm{U}_2) = \{[\Sigma_k \mathrm{Z}_k \mathrm{U}_2 \mathbf{z}] \in H_k(\mathcal{C}) \,|\, \mathbf{z} \in \mathbb{Z}^t\} \tag{3.23}$$

since the elements $\Sigma_k\mathrm{Z}_k\mathrm{U}_2$ do not belong to $\operatorname{im} i_k$. However, the elements $\Sigma_k\mathrm{Z}_k\mathrm{U}_2\mathrm{D}$ belong to $\operatorname{im} i_k$ and consequently $j_k(\Sigma_k\mathrm{Z}_k\mathrm{U}_2\mathrm{D}) = 0$ holds. The generators of the torsion subgroup of Z_k/B_k have a representations $\mathbf{T}_i$ which are the columns of the matrix $\mathrm{Z}_k\mathrm{U}_2$. Their corresponding torsion coefficients are $t_i = \mathrm{D}_i$.

In cohomology computation, let $\Sigma^k\mathrm{Z}^k$ be a basis for the group $Z^k(\mathcal{C};\mathbb{Z})$ and $\Sigma^k\mathrm{B}^k$ be a basis for the group $B^k(\mathcal{C};\mathbb{Z})$. To solve the quotient problem, examine the short exact sequence

$$0 \leftarrow B^k(\mathcal{C};\mathbb{Z}) \xleftarrow{\tilde{i}_k} Z^k(\mathcal{C};\mathbb{Z}) \xleftarrow{\tilde{j}_k} H^k(\mathcal{C};\mathbb{Z}) \leftarrow 0, \tag{3.24}$$

where the maps are defined by $\tilde{i}_k(\zeta) = \zeta \circ i_k$ and $\tilde{j}_k([\zeta]) = [\zeta] \circ j_k$ for $\zeta \in Z^k(\mathcal{C};\mathbb{Z})$. The sequence is exact, since

$$(\tilde{i}_k \circ \tilde{j}_k)([\zeta]) = \tilde{i}_k(\tilde{j}_k([\zeta])) = [\zeta] \circ j_k \circ i_k = [\zeta] \circ 0 = 0 \tag{3.25}$$

holds for all $[\zeta] \in H^k(\mathcal{C},\mathbb{Z})$. The surjection $\tilde{i}_k$ has an injective right-inverse $\tilde{i}_k^{-1}$ and the injection $\tilde{j}_k$ has a surjective left-inverse $\tilde{j}_k^{-1}$. Then, one has a short exact sequence

$$0 \to B^k(\mathcal{C};\mathbb{Z}) \xrightarrow{\tilde{i}_k^{-1}} Z^k(\mathcal{C};\mathbb{Z}) \xrightarrow{\tilde{j}_k^{-1}} H^k(\mathcal{C};\mathbb{Z}) \to 0. \tag{3.26}$$

From that sequence one can deduce representations for the generators of $H^k(\mathcal{C};\mathbb{Z})$ similarly as it was done in the homology computation.

To summarize, exactly the same computations were carried out in both the homology quotient problem and in the cohomology quotient problem. Let

$$0 \to B \xrightarrow{i} Z \xrightarrow{j} Z/B \to 0 \tag{3.27}$$

be a short exact sequence, where the group Z is finitely generated free abelian and B is its subgroup. The algorithm Quotient finds a basis representation H for the free subgroup F of the quotient group Z/B, representations $\mathbf{T}_i$ of generators of its torsion subgroups $\mathbf{T}_i$ and their corresponding torsion coefficients t_i, when matrices B and Z that represent bases for B and Z are given as an input.

Procedure Quotient

Input: Matrices Z and B
Output: A Matrix H, vectors $\mathbf{T}_i$ and integers t_j
Decompose $\mathrm{Z} = \mathrm{U} \begin{bmatrix} \mathrm{S_d} \\ 0 \end{bmatrix} \mathrm{V}$;
Let $\begin{bmatrix} \mathrm{T}_1 \\ \mathrm{T}_2 \end{bmatrix} := \mathrm{U}^{-1}$;
Let $\mathrm{I} = \mathrm{V}^{-1}{\mathrm{S_d}}^{-1}\mathrm{T}_1\mathrm{B}$;
Decompose $\mathrm{I} = \begin{bmatrix} \mathrm{U}_1 & \mathrm{U}_2 & \mathrm{U}_3 \end{bmatrix} \begin{bmatrix} \mathrm{I} & 0 \\ 0 & \mathrm{D} \\ 0 & 0 \end{bmatrix} \mathrm{V}$;
Let $\mathrm{H} = \mathrm{ZU}_3$;
Let $\mathbf{T}_i = (\mathrm{ZU}_2)_i \quad \forall\, i$;
Let $t_i = \mathrm{D}_i$;

3.3.4 The homology and cohomology computation algorithm

The final algorithm 3.3.1 to compute homology and cohomology of a chain complex $\mathcal{C}$ produces the representatives of the generators of the homology and cohomology groups. It joins together the algorithms for the kernel-image problem and for the quotient problem.

3.4 The homology and cohomology solver in *Gmsh*

We have implemented a homology and cohomology solver in the finite element mesh generator *Gmsh*. It knits together homology or cohomology reduction techniques presented in this thesis and consequently applies the homology or cohomology computation algorithm to the reduced chain complex.

The input for the solver is up to 3-dimensional finite element mesh that is generated by *Gmsh* or imported to *Gmsh*. The user designates a part of the mesh that is interpreted as the manifold M and a part that is interpreted as its subdomain S.

Algorithm 3.3.1: An algorithm for the homology and cohomology computation.

Input: Integer matrices D_k
Output: Integer matrices H_k and T_k and integer vectors $\mathbf{t}_k$
foreach ***k*** **do**
  if ***Compute homology*** **then**
    $\mathrm{D} := \mathrm{D}_k$;
    $q := k - 1$;
  else if ***Compute cohomology*** **then**
    $\mathrm{D} := \mathrm{D}_{k+1}^T$;
    $q := k + 1$;
  $(\mathrm{Z}_k, \mathrm{B}_q) :=$ `KerIm`(D);
foreach ***k*** **do**
  $(\mathrm{H}_k, \mathrm{T}_k, \mathbf{t}) :=$ `Quotient`$(\mathrm{Z}_k, \mathrm{B}_q)$;

The resulting representatives of the generators of the k-homology and k-cohomology groups are stored as sets of k-dimensional mesh elements that correspond to the non-zero k-cells of the representative k-chain. As k-chains are integer combinations of the mesh elements, the set contains a mesh element multiply times according to its integer coefficient and it is oriented according to the sign of the integer.

3.4.1 Homology computation routine

The routine for the homology computation is the following

1. Designate which part of the finite element mesh covers the manifold M and its submanifold S
2. Create a cell complex $K = (M, \mathcal{K}, \iota)$ with algorithm 3.1.1 from the finite element mesh
3. Remove cells that have been flagged to belong to the submanifold S
4. Run Omit-collapse algorithm 3.2.3
5. Run k-combine algorithm 3.2.2 followed by $k-1$-collapse algorithm 3.2.1 for each $k = \{n, \ldots, 1\}$.
6. Run the homology computation algorithm 3.3.1 to compute the homology groups of the chain complex $\mathcal{C}(M, S)$
7. Express the results in terms of the original finite element mesh

3.4.2 Cohomology computation routine

The routine for the cohomology computation is the following

1. Designate a manifold M and its submanifold S from a finite element mesh

2. Create a cell complex $K = (M, \mathcal{K}, \iota)$ with algorithm 3.1.1 from the finite element mesh

3. Remove cells that have been flagged to belong to the submanifold S

4. Run Coreduction algorithm 3.2.4

5. Run the dual version of the k-combine algorithm 3.2.2 followed by the dual version of the $k+1$-collapse algorithm 3.2.1 for each $k = \{0, \ldots, n-1r\}$

6. Run the cohomology computation algorithm 3.3.1 to compute the cohomology groups of the chain complex $\mathcal{C}(M, S)$

7. Express the results in terms of the original finite element mesh

3.5 Post-processing of homology and cohomology

The computational methods in the previous chapter produce representatives of generators of the homology and cohomology groups, once given a finite element mesh as an input.

These representations of the generating sets can further be processed since they are non-unique on two levels:

1. The generators of homology and cohomology groups are cosets of chains and cochains. Thus, there's an equivalence class of possible representatives for each generator in a generating set.

2. The generating set can be changed into an another by an unimodular transformation. This corresponds to a change of basis of a vector space.

The motive to perform such post-processing of the representations of the homology and cohomology groups is the topic of the next chapter, where homology and cohomology computations are applied in engineering problems in electromagnetics. In short, the generators are given a context-dependent meaning in engineering. In order to enable that, the engineer or an automated algorithm needs to be able to choose a representation that fits the purpose.

Starting from this section, we concentrate on the homology and cohomology *vector spaces* rather than groups. In the previous section, a computational method to find bases for the free subgroups of the homology and cohomology was described. Such bases are also bases for the homology and cohomology vector spaces.

The reason why this thesis first considered groups rather than vector spaces was two-fold:

1. To ensure that the computationally produced basis element representatives correspond to integer combinations of cells, rather than linear combinations which may have non-integer coefficients.

2. To recognize if the homology and cohomology groups had torsion subgroups. The existence of torsion in a finite element model is often an indicator some technical difficulty or human error, which might prevent its use in the finite element method [28, 61].

Once a basis for the free subgroup of homology or cohomology group has been obtained, it is a trivial matter of interpreting the integer coefficients of the representative chains as real or complex field coefficients to turn the basis into a basis of a vector space, see section 2.1.3 on page 27. However, in the following post-processing methods, the integer coefficients are retained.

3.5.1 Basis element representative selection

The basis element representative selection relies on that fact that if $z' = z + \partial_{k+1}c$ holds for some $c \in C_{k+1}$, the k-cycles z and z' in the chain complex $\mathcal{C}$ both represent the same element of the homology space $H_k(\mathcal{C})$. That is, one can always add a boundary to the basis element representative.

In a chain complex $\mathcal{C}(K)$ that is constructed from a cell complex K, one can examine whether the addition of a boundary k-cycle $\partial_{k+1}c$ reduces the norm $\|z\| := \langle z, z\rangle$ of the representative k-cycle. If $\|z'\| = \|z + \partial_{k+1}c\| < \|z\|$ holds, one can argue that the k-cycle z' is a simpler representative than z. It employs less k-cells or the same k-cells with lesser coefficients. Therefore, z' should be easier to comprehend visually.

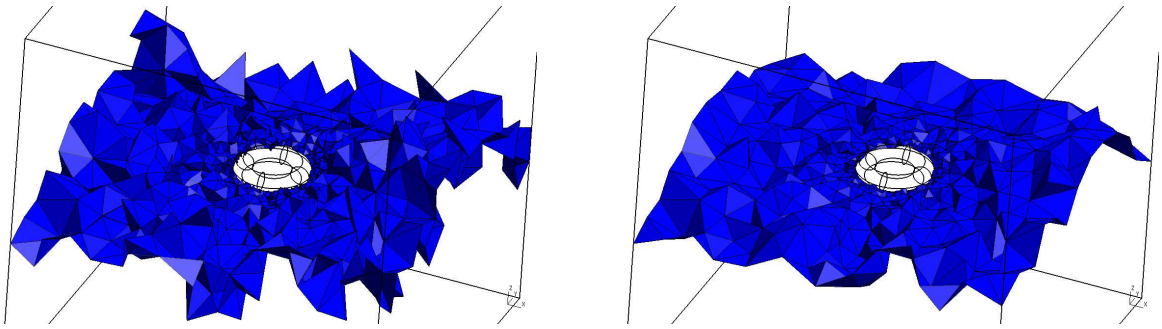


Figure 3.5: A representative of an element of $H_2(M, \partial M)$ before (1508 cells) and after (1003 cells) simplification by local addition of boundaries.

A computationally efficient method to find and examine suitable boundary k-cycle $\partial_{k+1}c$ candidates is to look for boundary k-cycles of single k-cells in the chain complex $\mathcal{C}(K)$. It is a greedy algorithm in a sense that it quickly somewhat reduces the norm while doesn't try to find the representative with the smallest norm. In Fig. 3.5 we have used such technique to simplify a representative of an element of $H_2(M, \partial M)$.

3.5.2 Basis selection

The homology and cohomology computation method presented in the previous chapter produces bases for the homology and cohomology spaces. The homology space $H_k(\mathcal{C}; \mathbb{F})$ basis $\{[z_1], [z_2], \ldots, [z_{\beta_k(\mathcal{C})}]\}$ is represented by a matrix H_k, whose column vectors $\mathbf{h}_i$ represent the basis element representatives $z_i = \Sigma_k \mathbf{h}_i$.

Since $H^k(\mathcal{C}; \mathbb{F})$ is a vector space, any invertible $\beta_k(\mathcal{C}) \times \beta_k(\mathcal{C})$ matrix A could be used as a change-of-basis matrix to produce a new basis AH^k. However, to ensure that the basis element representatives would still be integer combinations of the cells, the matrix A is

required to be unimodular. An analogous discussion holds for the representation matrix H_k of the cohomology space $H^k(\mathcal{C};\mathbb{F})$. The change of basis matrix A can be provided by the user, or it might be obtained by imposing some requirement on the basis. In the following sections, some methods to produce such unimodular change-of-basis matrices that the new basis has desirable properties are represented.

Homology basis and cohomology cobasis

If one has computed both k-homology and k-cohomology space bases of some chain complex, one might want them to compatible in the sense that there is a pairwise one-to-one correspondence between the elements of the bases. How to obtain such compatibility is discussed next.

Let $\mathrm{H}_k = \begin{bmatrix}\mathbf{h}_1 & \mathbf{h}_2 & \dots & \mathbf{h}_{\beta_k}\end{bmatrix}$ be basis representation matrix for a homology space $H_k(\mathcal{C})$ and let $\mathrm{H}^k = \begin{bmatrix}\mathbf{h}^1 & \mathbf{h}^2 & \dots & \mathbf{h}^{\beta_k}\end{bmatrix}$ be basis representation matrix for a cohomology space $H^k(\mathcal{C})$ obtained by the homology and cohomology computation. Then, one can automatically obtain a basis representation $\tilde{\mathrm{H}}^k = \begin{bmatrix}\tilde{\mathbf{h}}^1 & \tilde{\mathbf{h}}^2 & \dots & \tilde{\mathbf{h}}^{\beta_k}\end{bmatrix} = \mathrm{H}^k\mathrm{V}$ that satisfies

$$\mathrm{I} = \mathrm{H}_k^T\tilde{\mathrm{H}}^k = \mathrm{H}_k^T\mathrm{H}^k\mathrm{V} \iff \mathrm{V} = (\mathrm{H}_k^T\mathrm{H}^k)^{-1}, \tag{3.28}$$

where the change-of-basis matrix V is required to be unimodular. The matrix $\tilde{\mathrm{H}}^k$ is then the **cohomology cobasis** of the homology basis H_k, since

$$\mathbf{h}_i^T\tilde{\mathbf{h}}^j = [\tilde{z}^j]([z_i]) = \tilde{z}^j(z_i) = \delta_i^j, \quad 1 \le i,j \le \beta_k \tag{3.29}$$

holds for the bases. As an example, in Fig. 3.6 is basis representation for a homology space $H_1(M)$, and in Fig. 3.7 is its cohomology cobasis of $H^1(M)$.

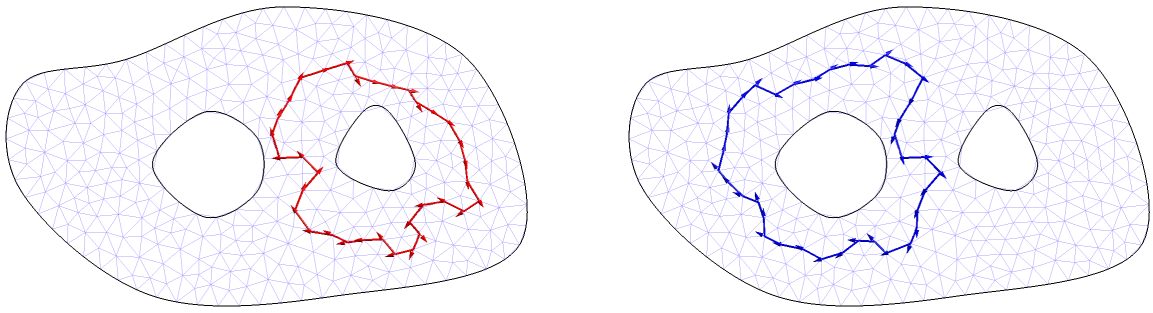


Figure 3.6: Basis representation for a homology space $H_1(M)$, where M is a planar domain with two holes.

Exact sequence based refinements

Exact sequences can be used to compute bases for vector spaces indirectly. For the computation of homology and cohomology space bases, Mayer-Vietoris sequence and the long exact homology sequence can be used. One can use them to deduce how to compute a basis for a homology or cohomology space by computing bases for spaces linked to it by an exact sequence.

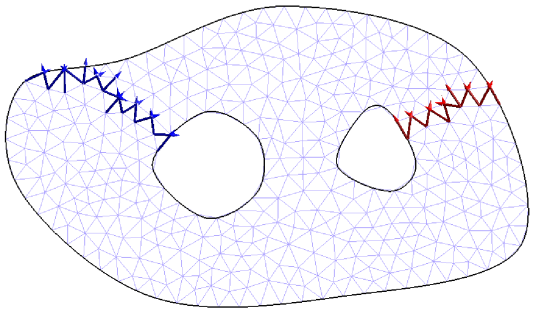


Figure 3.7: Basis representation for a cohomology space $H^1(M)$ which is the cobasis of the homology basis in Fig. 3.6.

If in a particular application some vector spaces in the sequence are trivial, the long exact sequences split into shorter ones, from which one can deduce alternative ways expressing the vector space of interest.

For a specific example, consider the following exact sequence of vector spaces

$$0 \xrightarrow{a} A \xrightarrow{b} B \xrightarrow{c} 0. \tag{3.30}$$

By the exactness of the of sequence, $\operatorname{im} a = \{0\} = \ker b$ and $\ker c = B = \operatorname{im} b$ hold, see section 2.1.1 on page 18. Therefore, the map b is an isomorphism of vector spaces. To compute a basis for B, one can first compute a basis for A and then map it by b to obtain a basis for B.

For a useful application to this, consider an n-manifold M, where $H^k(M) = H^{k+1}(M) = 0$ hold for $k < n$. Furthermore, let M have a decomposition $M = M_0 \cup M_1 \cup M_2$ so that $M_i \cap M_j = \partial M_i \cap \partial M_j$ hold when $i \neq j$. That is, the overlaps are $n-1$-manifolds.

By the Mayor-Vietoris sequence of cohomology [31], we have the exact sequence

$$0 = H^k(M) \to H^k(M_0 \cup M_1) \oplus H^k(M_0 \cup M_2) \to H^k(M_0) \to H^{k+1}(M) = 0. \tag{3.31}$$

Then, to compute a basis for $H^k(M_0)$, one can compute bases for $H^k(M_0 \cup M_1)$ and $H^k(M_0 \cup M_1)$, map them to $H^k(M_0)$ and concatenate the result to produce a basis for $H^k(M_0)$. This way, the basis of $H^k(M_0)$ reflects the decomposition $M = M_0 \cup M_1 \cup M_2$ of the whole manifold M. In the next chapter, we apply this to electromagnetic boundary value problems.

As an another example, consider the following exact sequence of vector spaces

$$0 \xrightarrow{a} A \xrightarrow{b} B \xrightarrow{c} C \xrightarrow{d} 0. \tag{3.32}$$

The short exact sequences of vector spaces always **split** [48], in contrast to the short exact sequences of abelian groups. It follows from the exactness of the sequence that b is an injection and c is a surjection. Therefore, $B = \operatorname{im} b \oplus \operatorname{im} c^{-1}$, where c^{-1} is the right-inverse of c. To compute a basis for B, one can thus compute bases for A and C instead and map the results B to form a basis of B.

3.5.3 Computation of harmonic representatives

An appealing visualization of the cohomology computation results is to compute the harmonic representatives of the basis elements of the cohomology spaces.

On an n-manifold M and a submanifold $S \subset \partial M$ the cohomology spaces $H^k(M,S)$ and de Rham cohomology spaces $H^k_{\mathrm{dR}}(M,S)$ are isomorphic via the Whitney map (2.126). Further, any element $[\omega] \in H^k_{\mathrm{dR}}(M,S)$ of de Rham cohomology space has a unique harmonic representative $\omega \in \mathcal{H}^k(M,S)$ by the isomorphism (2.117). One might argue that the harmonic representatives of de Rham cohomology spaces are visually pleasing and intuitive, since they often resemble a flow of liquid, for example.

In this section, a method to find harmonic representatives for the computed cohomology spaces are represented. As such, it is also a method to compute static problems in many fields of physics, such as electromagnetics, heat conduction, and the flow of incompressible and invicid fluid.

Harmonic k-forms cannot be exactly represented by Whitney k-forms. However, one can use the finite element method to find an approximate solution that satisfies the conditions $\mathrm{d}\omega = 0$ and $\mathrm{t}_S\omega = 0$ exactly, and the conditions $\mathrm{d}\star\omega = 0$ and $\mathrm{t}^*_S\star\omega = 0$ "weakly" [9, 10].

In section 2.4.4 on page 47 we ended up with the boundary value problem

$$\mathrm{d}\star\mathrm{d}\alpha = -\mathrm{d}\star\zeta, \tag{3.33}$$

$$\mathrm{t}_S\mathrm{d}\alpha = 0, \qquad \mathrm{t}_{S^*}\star\mathrm{d}\alpha = -\mathrm{t}_{S^*}\star\zeta, \tag{3.34}$$

for $\alpha \in H\Omega^{k-1}(M,S)$, where $\zeta \in H\Omega^k(M,S)$ is any representative of $[\zeta] \in H^k_{\mathrm{dR}}(M,S)$. If $\zeta = \sum_i \zeta_i \epsilon^i$ is a cochain representative of the cohomology group $H^k_{\mathrm{dR}}(M,S)$ obtained via cohomology computation, by Whitney map one obtains $\zeta = \sum_i \zeta_i \mathsf{w}^i_k$.

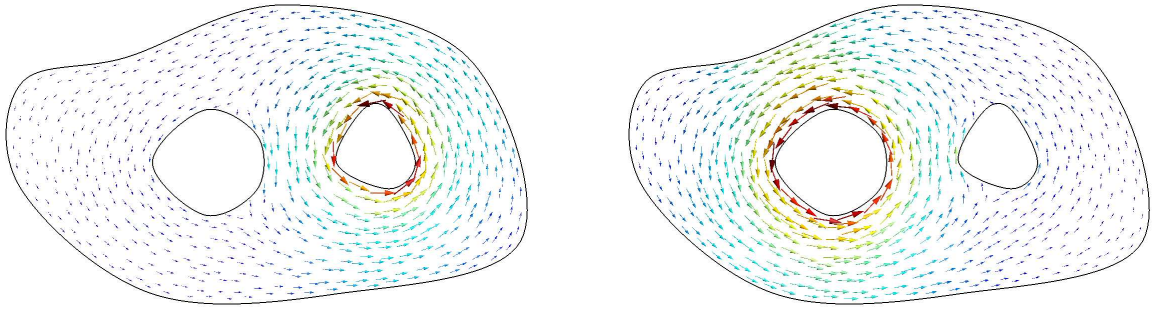


Figure 3.8: Harmonic representatives of the cohomology space $H^1(M)$ basis representation in Fig. 3.7.

From 3.33, one obtains the weak equation

$$(-1)^{n-k}\int_M \star\mathrm{d}\alpha \wedge \mathrm{d}\alpha' = -\int_M \star\zeta \wedge \mathrm{d}\alpha' \tag{3.35}$$

$$+\int_{\partial M} (\mathrm{t}\star\mathrm{d}\alpha + \mathrm{t}\star\zeta) \wedge \mathrm{t}\alpha' \quad \forall\, \alpha' \in H\Omega^{k-1}(M,S), \tag{3.36}$$

where the boundary integral vanishes since $\mathrm{t}_S\alpha' = 0$ and $\mathrm{t}_{S^*}\star\mathrm{d}\alpha = -\mathrm{t}_{S^*}\star\zeta$ hold. If one approximates $\alpha \in H\Omega^{k-1}(M,S)$ as an element of the Whitney space

$$\alpha = \sum_i \alpha_i \mathsf{w}^i_{k-1} \in W^{k-1}(M,S) \tag{3.37}$$

and also uses the basis of $W^{k-1}(M,S)$ as the test functions α', one obtains a linear system of equations

$$(-1)^{n-k}\sum_i \alpha_i \int_M \star\mathrm{d}\mathsf{w}^{k-1}_i \wedge \mathrm{d}\mathsf{w}^j_{k-1} = -\int_M \star\zeta \wedge \mathrm{d}\mathsf{w}^j_{k-1} \quad \forall\, j, \tag{3.38}$$

from which one can solve for $\boldsymbol{\alpha}$[1]. Finally, the approximate harmonic representative $\omega \in W^k(M,S)$ of the k-cochain $\zeta = \sum_i \zeta_i \epsilon^i$ is

$$\omega = \mathrm{d}\alpha + \zeta = \sum_i \alpha_i \mathrm{d}\mathsf{w}^i_{k-1} + \sum_i \zeta_i \mathsf{w}^i_k. \tag{3.39}$$

In Fig. 3.8 we represent the harmonic representatives of the cohomology space basis representation of Fig. 3.7 produced by the cohomology solver.

3.6 Examples

In this section, we present homology and cohomology computation examples together with benchmarks. The benchmarks use the homology or cohomology computation routines described in the section 3.4 on page 3.4 that are implemented in *Gmsh*.

3.6.1 Example: Solid cube

In this example, we exhibit bases of various (relative) homology and cohomology spaces one can compute to a cube. We also present benchmark results with different meshes of the cube.

We denote by M the three-manifold presenting the cube and by S two opposite facets of the cube and by $S^* = \partial M \setminus S$ their complement at the boundary ∂M. We compute bases of homology and cohomology spaces whose dimensions are presented in the Table 3.2. In this case, Lefschetz duality theorem evident. In figures 3.9 and 3.10 are

Table 3.2: Dimensions of a various homology and cohomology spaces of a cube.

	$(M,\emptyset)$	(M,S)	(M,S^*)	$(M,\partial M)$
β_0	1	0	0	0
β_1	0	1	0	0
β_2	0	0	1	0
β_3	0	0	0	1

[1]The choice to represent the cochain ζ in terms of Whitney elements results the above *Galerkin Hodge* [10] of cochains. A different choice would lead to a different approxiamation of the harmonic representative.

the representatives of the basis elements of the homology and cohomology spaces. The homology and cohomology computation times are listed in the Table 3.3. In the Table 3.3 we have also approximated the polynomial computational complexity $O(n^\alpha)$ of the computations, where n is the number of tetrahedra of the mesh.

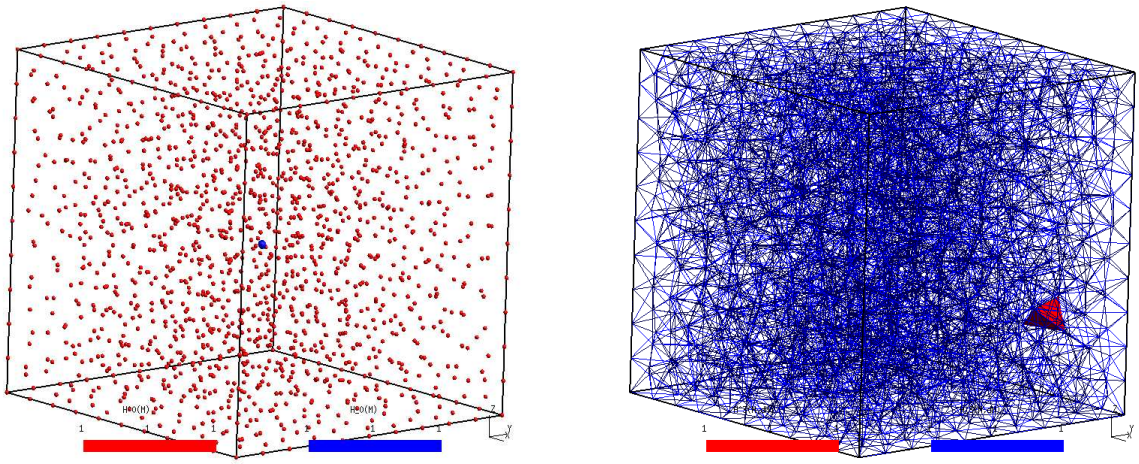


Figure 3.9: On the left are the basis element representatives of the spaces $H_0(M)$ (blue node) and $H^0(M)$ (red nodes). Similarly, on the right they are for $H_3(M, \partial M)$ (blue tetrahedra) and $H^3(M, \partial M)$ (red tetrahedron). All the four vector spaces are isomorphic.

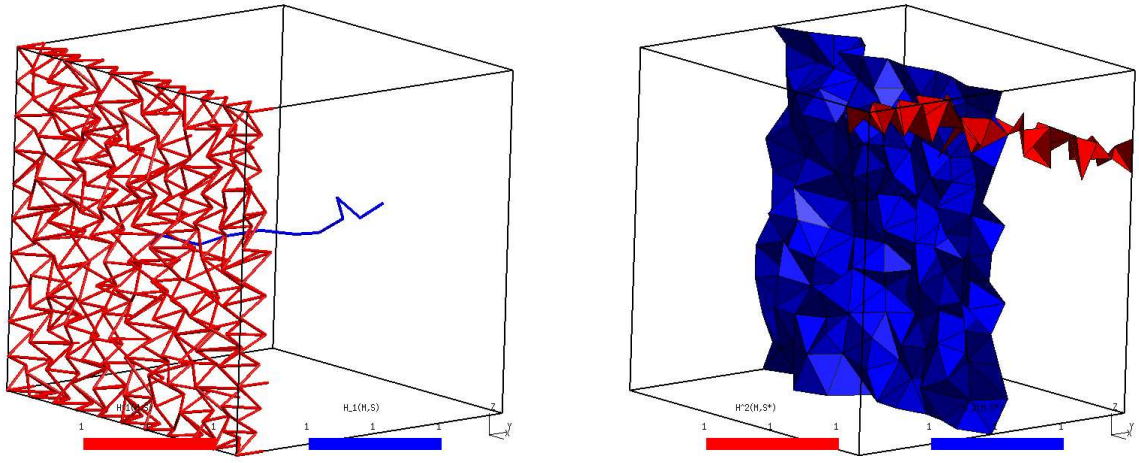


Figure 3.10: On the left are the basis element representatives of the spaces $H_1(M, S)$ (blue edges) and $H^1(M, S)$ (red edges). Similarly, on the right they are for $H_2(M, S^*)$ (blue triangles) and $H^2(M, S^*)$ (red triangles). All the four vector spaces are isomorphic.

3.6.2 Example: Closed surfaces

By closed surfaces we mean compact and closed 2-manifolds that are embedded to $\mathbb{R}^3$. Such manifolds are necessarily orientable, since they divide $\mathbb{R}^3$ in two components: the

Table 3.3: Benchmarks of the homology and cohomology computation for the cube. In this example, homology and cohomology computations have slightly higher computational complexity $O\big(()n^{\alpha}\big)$ than Delaunay mesh generation algorithm [30] in *Gmsh*.

	Mesh 1	Mesh 2	Mesh 3	Mesh 4	α
#Tetrahedra, n	64 807	321 415	557 086	1 077 478	
Mesh time [s]	1.0	5.4	9.7	19	1.05
$H_0(M)$	2.6	17	32	67	1.16
$H^0(M)$	3.0	18	34	67	1.11
$H_3(M, \partial M)$	3.7	27	52	119	1.23
$H^3(M, \partial M)$	2.7	16	31	65	1.13
$H_1(M, S)$	2.8	17	34	72	1.16
$H^1(M, S)$	3.2	19	36	83	1.15
$H_2(M, S^*)$	3.0	20	39	89	1.20
$H^2(M, S^*)$	2.6	17	31	65	1.15

inside and the outside of the surface. The surface is the boundary of the inside component and can inherit its orientation.

The k**-collapse** algorithm and its dual version are unable to reduce the chain complex constructed from a closed surface. However, since the such surfaces are orientable, one can safely use Omit-collapse and Coreduction algorithms, which relied on the Lefschetz duality that required the manifold to be orientable. With these algorithms in service, we can expect the (co)homology computation of closed surfaces to be very efficient.

We compare the 2D mesh generation and homology and cohomology computation times for closed surface S with two handles. We have depicted the surface together with generators for the groups $H_1(S)$ and $H^1(S)$ in Fig. 3.11. The benchmark results are in the Table 3.4.

Table 3.4: Benchmarks of the homology and cohomology computation of a closed surface S. Surprisingly, cohomology computation is done faster than the mesh generation.

	Mesh 1	Mesh 2	Mesh 3	Mesh 4	α
#Triangles	53 340	212 426	480 226	983 170	
Mesh time [s]	1.7	8.3	24.3	53.0	1.19
$H_1(S)$	1.8	11.0	33.2	87.0	1.33
$H^1(S)$	1.1	5.4	15.9	34.7	1.20

3.6.3 Example: Torus knots

Knot is an embedding of an abstract circle S^1, i.e. the only closed, compact, and connected 1-manifold, to $\mathbb{R}^3$. A (p, q) **torus knot** is a knot that is imagined to be drawn on a surface of a torus in $\mathbb{R}^3$. The *coprime* integers p and q indicate how many times the knot winds around the major and the minor radius of the torus, respectively. For $p = 2$ and $q = 3$ the knot is called the *trefoil knot*.

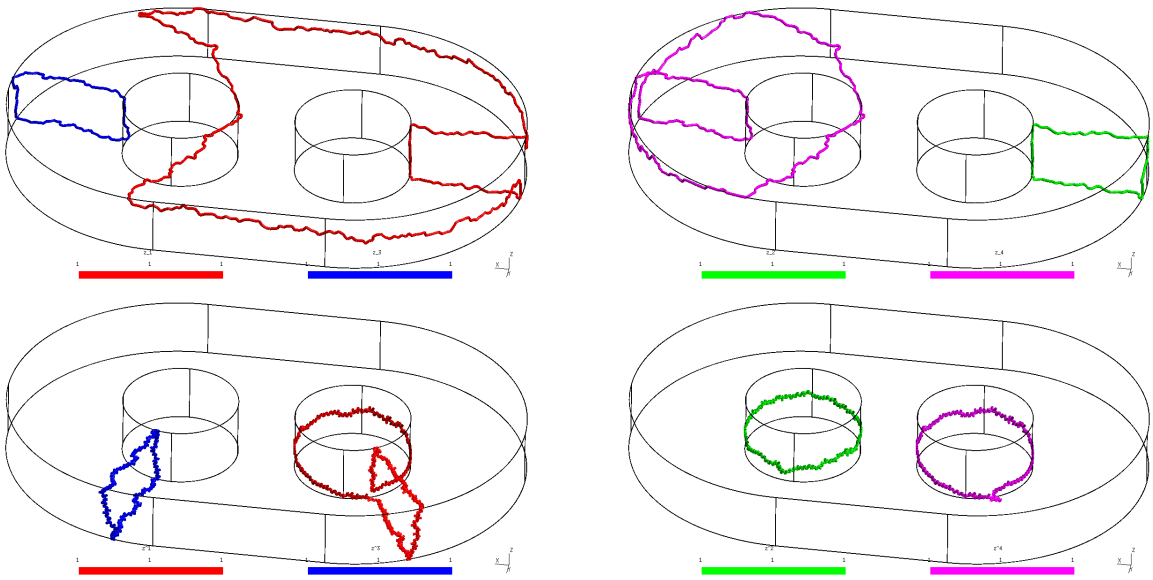


Figure 3.11: A closed surface S with two handles. On the top row are the four generators of the group $H_1(S)$. On the bottom row are the four generators of the group $H^1(S)$.

We consider a tube M_c in a solid cube $M \subset \mathbb{R}^3$ whose centroid is a torus knot and compute the homology space $H_2(M_a, \partial M_a)$ and the cohomology space $H^1(M_a)$ where $M_a = M \setminus M_c$ is the complement of the tube M_c. Such is a classical benchmark for computing "cuts" for the magnetic scalar potential. A representative of the basis element of $H_2(M_a, \partial M_a)$ is called a thin cut and a representative of the basis element of $H^1(M_a)$ is usually called a thick cut [41, 21].

Knotted geometries like this have been of interest since the early days of electromagnetics and have found applications to produce so-called force-free magnetic fields [13]. They have also helped to clarify the distinction between homology and homotopy.

In Table 3.5 we compare homology and cohomology computation times for various torus knots with approximately the same number of mesh elements. Notice the poor performance of the $(11, 2)$ torus knot. In that case, the (co)homology reduction algorithms perform poorly. In figures 3.12, 3.13, and 3.14 are the corresponding representatives of the basis elements of $H_2(M_a, \partial M_a)$ and $H^1(M_a)$ for the knots.

Table 3.5: Benchmarks of the homology and cohomology computation of complements of torus knots. In the case $p = 11, q = 2$, the reduction algorithms are found to perform poorly.

Torus knot	$p = 2, q = 3$	$p = 3, q = 31$	$p = 11, q = 2$
#Tetrahedra	248 561	250 876	243 051
Mesh time [s]	5.9	5.9	6.3
$H_2(M_a, \partial M_a)$	16	17	346
$H^1(M_a)$	19	19	480

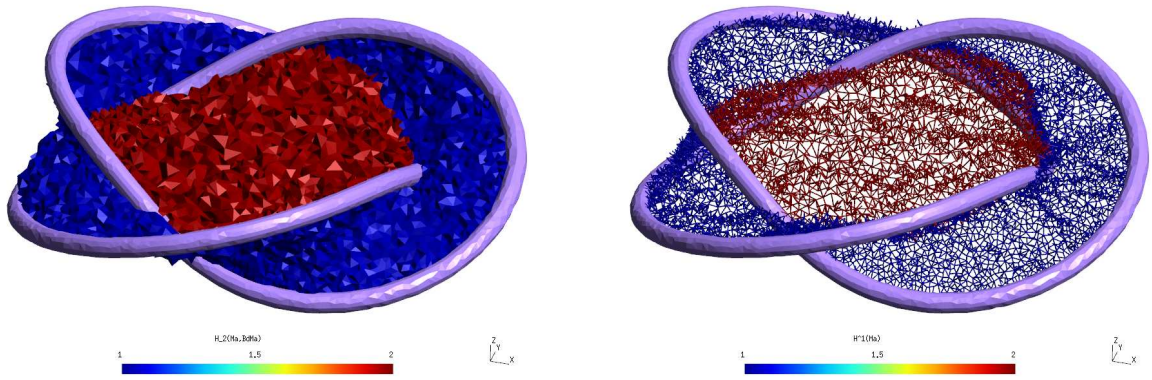


Figure 3.12: $(2, 3)$ torus knot. On the left is a basis element representative of the space $H_2(M_a, \partial M_a)$. On the right is a basis element representative of the space $H^1(M_a)$.

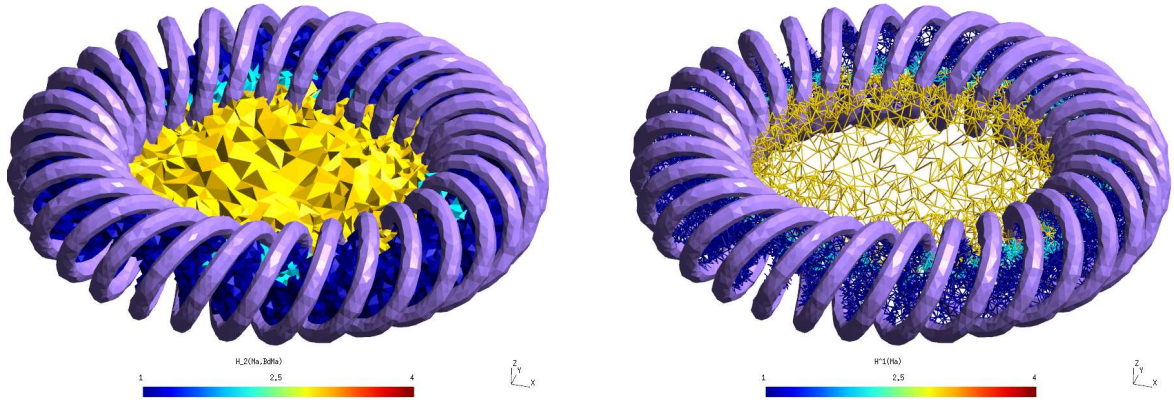


Figure 3.13: $(3, 31)$ torus knot. On the left is a basis element representative of the space $H_2(M_a, \partial M_a)$. On the right is a basis element representative of the space $H^1(M_a)$.

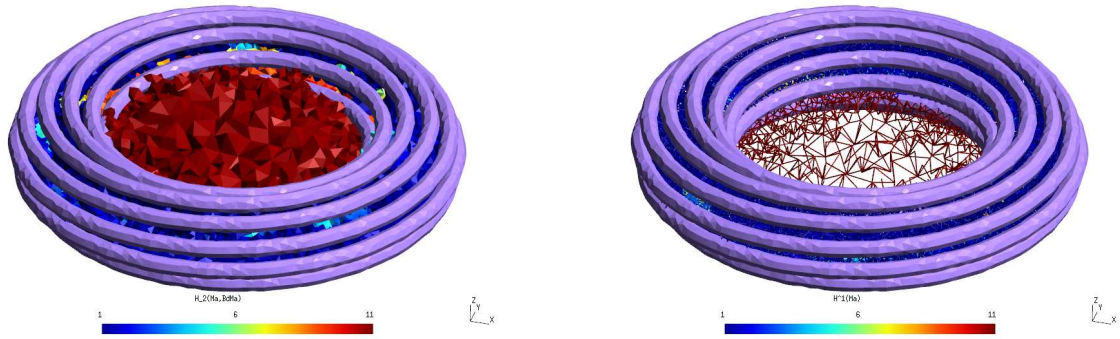


Figure 3.14: $(11, 2)$ torus knot. On the left is a basis element representative of the space $H_2(M_a, \partial M_a)$. On the right is a basis element representative of the space $H^2(M_a)$.

Chapter 4

Application of cohomology in the finite element method for electromagnetics

In electrical engineering, field problems are often coupled to circuit problems. Then, it is convenient to drive a static or quasistatic field problem with net quantities, that serve as the state variables of the circuit. For example, such net quantities can be a voltage difference between terminals or a net current through a terminal. Or, there may be a specific relation between the current and the voltage which is specified by the circuit problem.

In such situation to drive the problem with specific electric field or current density distribution is not justified, as the information of how they are distributed is not provided by the circuit model. Instead, one assumes that the terminals are perfect electric conductors and that elsewhere there is a perfect insulation of electric current.

The perceived difficulty to drive an electromagnetic field problem with net quantities depends on the chosen potential formulation. For example, in static electrokinetic problem with electric scalar potential formulation it is almost imminent how to drive the problem with a voltage difference. The method to drive it with current is bit more tedious to implement, and the electric vector potential formulation takes the issue to a whole new level.

At first sight, it seems that each formulation and each driving quantity has its own set of tricks that one can use. However, differential forms and their relation to the homology and the cohomology are able to provide a general framework where the symmetry between different potential formulations and drive quantities is clearly exposed.

In this chapter, it is described how electromagnetic boundary value problems can be formulated and solved by exploiting the homology and cohomology computation, and their link to the theory of harmonic forms. Similar formulations, with an emphasis on the eddy current problems, have been considered in a wealth of publications [15, 41, 17, 16, 40, 55, 33].

4.1 Electromagnetic modeling

The classical electromagnetic theory is summarized by Maxwell's equations in the Euclidean 3-space $\mathbb{E}^3$:

$$\int_{\partial\Sigma} e = -\partial_t \int_{\Sigma} b \quad \forall\, \Sigma, \qquad \int_{\partial\Omega} b = 0 \quad \forall\, \Omega, \tag{4.1}$$

$$\int_{\partial\Sigma} h = \int_{\Sigma} j + \partial_t \int_{\Sigma} d \quad \forall\, \Sigma, \qquad \int_{\partial\Omega} d = \int_{\Omega} \rho \quad \forall\, \Omega, \tag{4.2}$$

$$\tag{4.3}$$

and pointwise constitutive equations

$$d = \star\epsilon e, \qquad b = \star\mu h, \qquad j = \star\sigma e. \tag{4.4}$$

The differential 1-forms e and h are called electric field and magnetic field intensity, respectively. The differential 2-forms b, d, and j are called magnetic flux density, electric flux density and current density, respectively. The differential 3-form ρ is called electric charge density. The material operators ϵ, μ, and σ that map 1-forms to 1-forms are called permittivity, permeability, and conductivity, respectively. The surface Σ and the volume Ω are any 2-dimensional and 3-dimensional submanifolds of $\mathbb{E}^3$.

In electromagnetic modeling, one is often not concerned about the electromagnetic fields in all $\mathbb{E}^3$. Instead, one often chooses an submanifold $M \subset \mathbb{E}^3$, with boundary ∂M on which to set up a boundary value problem. By the generalized Stokes' theorem, the Maxwell's integral equations can be converted into partial differential equations in M. For modeling purposes, the manifold M does not need to be explicitly a subset of the Euclidean 3-space, but an oriented compact Riemannian manifold (M, g) with any metric tensor g.

Also, the dimension of M may be less than 3 based on arguments about the symmetries of the problem. However in this chapter, for the sake of clarity and to exploit the intuition of an electrical engineer, we set the dimension of M equal to 3.

The price to pay for the truncation of the modeling domain is to define the boundary conditions and the emergence of cohomology in the problem domain. In a large class of electromagnetic boundary value problems, the unknown field, or a component of it, is a harmonic field. Harmonic 1-, and 2-forms on a 3-manifold correspond to vector fields that are both irrotational and solenoidal. In addition, on complementary parts of the boundary of the domain, the tangential and the normal components of a irrotational and solenoidal vector field vanish.

Since the space of harmonic forms is isomorphic to de Rham cohomology space, the unknown field is not uniquely determined until one has fixed a number of coefficients that is equal to the dimension of de Rham cohomology space. That is, one has to choose a solution from a finite dimensional vector space. ***We call the choice an element from de Rham cohomology space the cohomology condition of the boundary value problem.***

This chapter describes how cohomology conditions can be imposed in a large class of electromagnetic boundary value problems. Specifically, we exploit the homology and cohomology computation results to construct *cohomology basis functions* for the finite element method.

4.2 Static problems

In this section, we examine finite element element formulations for static electromagnetic boundary value problems with cohomology conditions. We use the static problems as an introduction to the cohomological concepts, and as a stepping stone when we discuss about eddy current problems in the next section.

We introduce *cohomology basis functions* as means to fix cohomology conditions of the boundary value problems in the finite element method. Cohomology basis functions are obtained via cohomology computation, while some care needs to put in to obtain such cohomology basis functions that fix the desired cohomology conditions. We will find their use in the eddy current problems to be very beneficial. However, the use of cohomology basis functions in statics has some intrinsic value regarding electrical circuits.

From static electromagnetic problems, one can extract at post-processing stage lumped parameters used in time-invariant linear electrical circuits: resistance, capacitance, and inductance. The coefficients of the cohomology basis functions correspond to currents, voltages, electric and magnetic fluxes, ratios of which are the lumped parameters. That is, the coefficients can be used to drive the problems with net quantities, and the extraction of a lumped parameter simplify from integration of the dissipated or stored energy to a computation of a ratio.

4.2.1 Electrokinetic problem

In the electrokinetic problem, one is to find a pair of harmonic forms $(e, j) \in \mathcal{H}^1(M, S_e) \times \mathcal{H}^2(M, S_j)$ on the 3-manifold M, where $S_e \subset \partial M$ and $S_j = \partial M \setminus S_e$ hold. In other words

$$\mathrm{d}e = 0, \qquad \mathrm{d}j = 0, \qquad j = \sigma \star e, \tag{4.5}$$

$$\mathrm{t}_{S_e} e = 0, \qquad \mathrm{t}_{S_j} j = 0 \tag{4.6}$$

hold for the electric field e and the current density j. Fig. 4.1 represents an example 3-manifold M with submanifolds S_e and S_j.

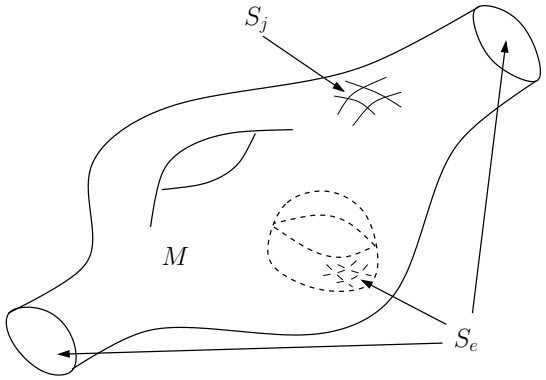


Figure 4.1: An example 3-manifold M that can be a domain of an electrokinetic problem. It has a void and a tunnel through it. The condition $\mathrm{t}e = 0$ on the boundary of the void is unusual in electromagnetic modeling, but may have some modeling purposes. This possibility is included here for generality.

The manifold M has non-trivial cohomology spaces modulo S_e and S_j. Therefore, one has to choose the cohomology class of the solution before one can obtain a unique solution to the problem. That is, the following cohomology conditions need to be set. Let $\{z_i^1\}$ and $\{z_i^2\}$ form bases for $H_1(M, S_e)$ and $H_2(M, S_j)$, respectively.

$$\int_{z_i^1} e = V_i \quad \forall\, 1 \leq i \leq \beta_1(M, S_e) \quad \text{and} \tag{4.7}$$

$$\int_{z_i^2} j = I_i \quad \forall\, 1 \leq i \leq \beta_2(M, S_j), \tag{4.8}$$

of which only N of them, $N = \beta_1(M, S_e) = \beta_2(M, S_j)$, are independent, since the Hodge operator is an isomorphism between the spaces $\mathcal{H}^1(M, S_e)$ and $\mathcal{H}^2(M, S_j)$.

The cohomology conditions can be interpreted as electromotive forces along paths and as total currents through surfaces. Actually, each condition enforces the electromotive force or total current to a class of paths or surfaces that are homologous. Representatives of such paths and surfaces for the example manifold M in the Fig. 4.1 are exhibited the Fig. 4.2. The bases for the spaces $H_1(M, S_e)$ and $H_2(M, S_j)$ in the figure are such that the basis elements are in such one-to-one correspondence that each path z_i^1 pierces the corresponding surface z_i^2 exactly once and pierces no other surfaces.

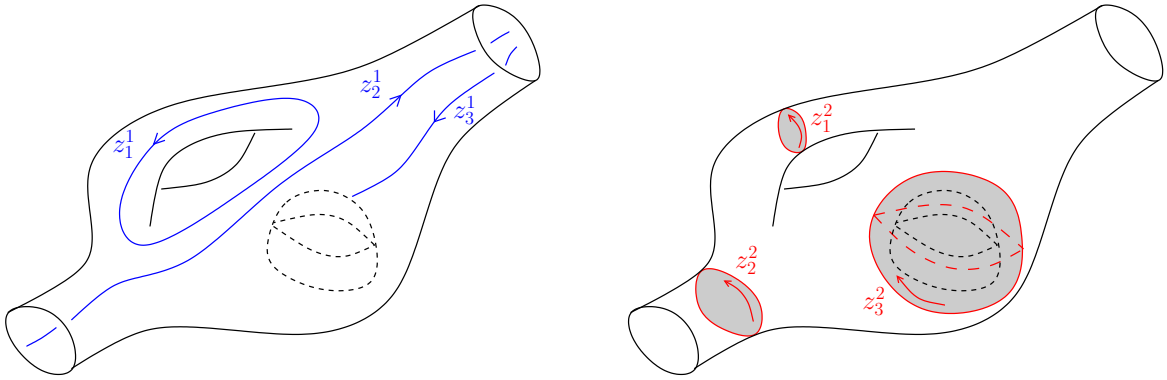


Figure 4.2: Representatives of the basis elements of the homology spaces $H_1(M, S_e)$ and $H_2(M, S_j)$ that are used in the cohomology conditions of the electrokinetic problem. The (relative) 1-cycles z_i^1 represent the paths along which the voltage $V_i = \int_{z_i^1} e$ is determined. The (relative) 2-cycles z_i^2 represent the surfaces through which the net current $I_i = \int_{z_i^2} j$ is determined. Since $\mathrm{d}e = 0$ and $\mathrm{d}j = 0$ hold in M, the voltage or current is independent of the choice of representative from the homology classes $[z_i^1]$ and $[z_i^2]$.

The cohomology conditions are specified using a basis of the homology space $H_1(M, S_e)$. Any basis will do, but in engineering applications, some bases are more preferable than others. For example in the electrokinetic problem, a path that spans from a terminal to a terminal once, like z_2^1 in Fig. 4.2, rather than multiple times or via some third terminal, might be the most useful.

In the finite element method, the cohomology conditions can be fixed using so-called cohomology basis functions, which we will define next.

4.2.2 Cohomology basis functions

Given a finite element mesh of a manifold M, the elements of de Rham cohomology space $H^k_{\mathrm{dR}}(M,S)$ can be represented using the Whitney forms. To be able to fix the cohomology conditions of boundary value problem in a unified manner, we will define *cohomology basis functions* as integer combinations of Whitney forms for the task.

In the finite element method, the ordinary basis functions have associated coefficients, some of which are fixed in advance as Dirichlet conditions and some of which are degrees of freedom of the problem.

Similarly to the cohomology basis functions we associate coefficients that can be interpreted as net quantities such as current or voltage, i.e. as the cohomology conditions. Ratios of the coefficients can be used to determine lumped parameters, for example the resistance.

The coefficients can either be fixed in advance, or be a part of the finite element solution of the problem. Thus in the formulations, as we shall see, they behave much like Dirichlet and Neumann boundary conditions in the conventional finite element method for scalar fields.

Definition and properties

Let M be a 3-manifold and $S \subset \partial M$ its 2-submanifold. Let $\zeta^i = \sum_j \zeta^i_j \epsilon^j \in C^k(M,S)$ (see section 2.3.2 on page 40) be a k-cocycle so that $[\zeta^i] \in H^k(M,S)$ holds. We define **Cohomology basis function** W^i_k associated with ζ^i as

$$\mathsf{W}^i_k := w(\zeta^i) = \sum_j \zeta^i_j \mathsf{w}^j_k \in W^k(M,S). \tag{4.9}$$

That is, to construct cohomology basis functions one has to compute a basis of $H^k(M,S)$ and interpolate the basis elements by the Whitney map w of section 2.4.5 on page 48.

Usually, the integer coefficient vector $\boldsymbol{\zeta}^i$ produced by the cohomology solver is sparse, i.e. the cohomology generators have a limited support within the manifold. Therefore, the use of cohomology basis functions in the finite element method does not fill the system matrix dramatically. Rather, each of them produces a dense block with a couple of dense rows and columns.

The cohomology basis functions span a subspace of $W^k(M,S)$ that we denote by

$$HW^k(M,S) := \{\mathsf{W}_k \in W^k(M,S) \,|\, [\mathsf{W}_k] \in H^k_{\mathrm{dR}}(M,S)\}. \tag{4.10}$$

Now, if $\{[\zeta^1],[\zeta^2],\ldots,[\zeta^{\beta_1(M,S)}]\}$ forms a basis for the cohomology space $H^k(M,S)$, the cosets of cohomology basis functions $\{[\mathsf{W}^1_k],[\mathsf{W}^2_k],\ldots,[\mathsf{W}^{\beta_k(M,S)}_k]\}$ form a basis for de Rham cohomology space $H^k_{\mathrm{dR}}(M,S)$. If $\{[\zeta^1],[\zeta^2],\ldots,[\zeta^{\beta_k(M,S)}]\}$ was a cobasis of a basis $\{[z_1],[z_2],\ldots,[z_{\beta_j(M,S)}]\}$ of $H_k(M,S)$, the associated de Rham cohomology basis functions satisfy

$$\int_{z_i} \mathsf{W}^j_k = \delta^j_i \quad \forall\, 1 \le i,j \le \beta_k(M,S), \tag{4.11}$$

i.e. they form de Rham cohomology cobasis for the basis of $H_k(M,S)$.

For clarity, later in this chapter we shall denote Whitney 0-, 1-, and 2-forms with n, e, and f, respectively, and cohomology basis functions with N, E, and F, respectively.

Selection of cohomology basis functions

If the dimensions of the homology and cohomology spaces are greater than one, it might be difficult to perceive to which cohomology condition a cohomology basis function is related to. In this section, we discuss how the user can be in control of the cohomology basis functions.

The first method is to separately compute bases for $H_k(M, S)$ and $H^k(M, S)$. Then, one can manually take integer combinations of the computed basis of $H_k(M, S)$ and then transform the basis of $H^k(M, S)$ to be its cobasis, as discussed in section 3.5.2 on page 3.5.2.

The second method utilizes the property of the long exact homology sequence that if $[z] \in H_k(M, S)$ holds, then $[\partial z] \in H_{k-1}(S)$ holds. Further, if $\beta_k(M) = 0$ holds, $[\partial z]$ is a non-zero element of $H_{k-1}(S)$.

For example, suppose M is a 3-manifold and let $\beta_1(M) = 0$ hold. Let M_c and M_a be conducting and non-conducting 3-submanifolds of M, respectively. Let $S = \partial M_a \cap \partial M_c$ be their interface, see Fig. 4.3. One wants to obtain a basis of $H_1(M_a)$ whose representative 1-cycles loop once around one conductor only. To obtain such basis one can compute the basis of $H_2(M_c, S)$ and compute the boundaries of the representative relative 2-cycles. The result is a basis for $H_1(S)$ and the basis elements are homologous to representatives of elements of a basis for $H_1(M_a)$. After obtaining the desirable basis of $H_1(M_a)$ one can transform a computed basis of $H^1(M_a)$ to be its cobasis.

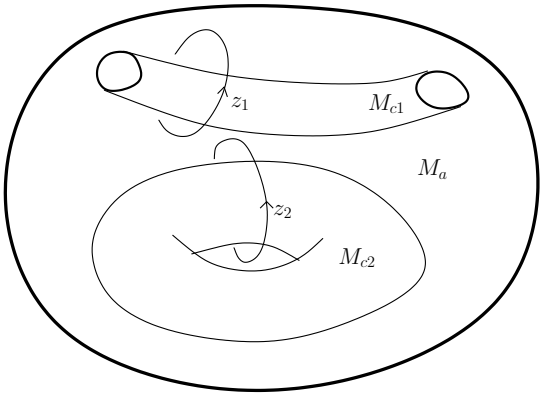


Figure 4.3: How to find a basis $\{[z_1, z_2]\}$ of $H_1(M_a)$ whose elements correspond to individual conductors in M_c.

The third method utilizes the Mayer-Vietoris sequence to obtain a desirable cohomology basis directly. Let again 3-manifold M have a decomposition as in the Fig. 4.3, and denote by M_{c1} and M_{c2} the components of the conducting submanifold M_c. If $\beta_1(M) = \beta_2(M) = 0$ hold, one may write the Mayer-Vietoris sequence for cohomology as

$$0 = H^1(M) \to H^1(M_a) \to H^1(M_a \cup M_{c1}) \oplus H^1(M_a \cup M_{c2}) \to H^2(M) = 0. \qquad (4.12)$$

That is, the spaces $H^1(M_a)$ and $H^1(M_a \cup M_{c1}) \oplus H^1(M_a \cup M_{c2})$ are isomorphic as described in the section 3.5.2 on page 71. Consequently, one can compute the bases $\{[\zeta^1]\}$ for $H^1(M_a \cup M_{c1})$ and $\{[\zeta^2]\}$ for $H^1(M_a \cup M_{c1})$ separately. Then, $\{[\tilde{i}\zeta^1], [\tilde{i}\zeta^2]\}$ is a basis for

the space $H^1(M_a)$, where i is the corresponding inclusion map from M_a to $M_a \cup M_{c1}$ or $M_a \cup M_{c1}$. In such basis, each basis element corresponds to just one conductor. By induction, this generalizes to any finite number of conductors in M.

4.2.3 Electric field -conforming formulation

In the electric field conforming formulation, we seek to satisfy $\mathrm{d}e = 0$ and $t_{S_e} e = 0$ exactly, and $\mathrm{d}j = 0$ and $\mathrm{t}_{S_j} j = 0$ in the weak sense. That is, the sought solution is such that $[e] \in H^1_{\mathrm{dR}}(M, S_e)$ holds. The conditions $\mathrm{d}e = 0$ and $t_{S_e} e = 0$ are satisfied by

$$e = \sum_{\sigma_i^0 \in M \setminus S_e} \varphi_i \mathrm{d}\mathsf{n}^i + \sum_{i=1}^{\beta_1(M,S_e)} V_i \mathsf{E}^i \in W^1(M, S_e), \tag{4.13}$$

where the cohomology basis functions are of the form $\mathsf{E}^i \in HW^1(M, S_e)$, i.e. $[\mathsf{E}^i]$ span $H^1_{\mathrm{dR}}(M, S_e)$.The support of the nodal basis functions n does not cover S_e to not fix the cohomology condition with them, but to leave the task completely for the cohomology basis functions.

This approximation for e deviates from the conventional finite element method, where e fully given in terms of Whitney 0-forms. However, our choice allows one to assign more general cohomology conditions on e when M contains tunnels. Such cohomology conditions correspond to induced electromotive forces around the tunnel, or circuit terminals that are not otherwise apparent in the finite element model.

Now, if one requires that

$$\int_M \star\sigma e \wedge e' = \begin{cases} 0 & \text{when } e' = \mathrm{d}\mathsf{n}^i \\ I_i & \text{when } e' = \mathsf{E}^i \end{cases} \quad \forall\, e' \in \{\mathrm{d}\mathsf{n}^i, \mathsf{E}^i\} \tag{4.14}$$

holds, the j-side conditions in (4.5) and (4.6), $\mathrm{d}j = \mathrm{d}\star\sigma e = 0$ and $\mathrm{t}_{S_j} j = \mathrm{t}_{S_j} \star\sigma e = 0$, are satisfied weakly, since

$$0 = \int_M \star\sigma e \wedge \mathrm{d}\mathsf{n}^i = \int_M \mathrm{d}\star\, \sigma e \wedge \mathsf{n}^i - \int_{S_j} \mathrm{t}\star\sigma e \wedge \mathrm{t}\mathsf{n}^i \quad \forall\, \mathsf{n}^i \tag{4.15}$$

holds. By plugging (4.13) into (4.14), one obtains a linear system with blocks

$$\begin{bmatrix} \int_M \star\sigma \mathrm{d}\mathsf{n} \wedge \mathrm{d}\mathsf{n} & \int_M \star\sigma \mathsf{E} \wedge \mathrm{d}\mathsf{n} \\ \int_M \star\sigma \mathrm{d}\mathsf{n} \wedge \mathsf{E} & \int_M \star\sigma \mathsf{E} \wedge \mathsf{E} \end{bmatrix} \begin{bmatrix} \boldsymbol{\varphi} \\ \mathbf{V} \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \mathbf{I} \end{bmatrix}. \tag{4.16}$$

From this it is evident that the system matrix is symmetric and positive-definite. If one fixes all the coefficients V_i, the system reduces to

$$\left[\int_M \star\sigma \mathrm{d}\mathsf{n} \wedge \mathrm{d}\mathsf{n}\right] \left[\boldsymbol{\varphi}\right] = -\left[\int_M \star\sigma \mathsf{E} \wedge \mathrm{d}\mathsf{n}\right] \left[\mathbf{V}\right]. \tag{4.17}$$

From the solution to this linear system, one obtains an approximation of $e \in \mathcal{H}^1(M, S_e)$ that satisfies the voltage cohomology conditions in (4.7) for the chosen cohomology basis functions E^i. The currents $\mathbf{I}$ can then be computed from the equation (4.16).

On the other hand, if one fixes all the coefficients I_i, the equation (4.14) reads

$$I_i = \int_M \star\sigma e \wedge \mathsf{E}^i = \int_M j \wedge \mathsf{E}^i = \int_{z_i^2} j \quad \forall\, 1 \leq i \leq \beta_2(M, S_j),\ [z_i^2] \in H_2(M, S_j) \tag{4.18}$$

where the last equality holds by the properties of *cup product* of cohomology [48]. That is, the current cohomology conditions in (4.7) are satisfied.

In the general case, but assuming M is connected, one must fix a total of N voltages and currents to obtain a unique solution to the electrokinetic problem (4.5) - (4.7), where $N = \beta_1(M, S_e) = \beta_2(M, S_j)$ hold. Otherwise one ends up with under- or over-determined system of equations. If M is not connected, the above statement hold for each connected component of M separately.

If $\beta_1(M) = 0$ holds, this formulation reduces to the "traditional" scalar finite element method. To see this, let $[z_1] \in H_1(M, S_e)$ hold. By the long exact homology sequence, $0 \neq \partial_*[z_1] = [z_0] \in H_0(S_e)$ holds. Then, the approximate solution of $e \in \mathcal{H}^1(M, S_e)$ can be expressed as an exterior derivative of a scalar potential

$$\varphi = \sum_{\sigma_i^0 \in M \setminus S_e} \varphi_i \mathsf{n}^i + \sum_{i=1}^{\beta_1(S_e)-\beta_0(M)} V_i \mathsf{N}^i \in W^0(M), \tag{4.19}$$

where the cohomology basis functions $\mathsf{N}^i \in W^0(M)$ and satisfy $[\mathrm{t}\mathsf{N}^i] \in H^0_{\mathrm{dR}}(S_e)$. In this case, not the full basis of $H^0_{\mathrm{dR}}(S_e)$ is used, but just such basis functions that have non-zero image in $H^1_{\mathrm{dR}}(M, S_e)$. Since $\mathrm{t}\mathrm{d}\mathsf{N}^i = 0$ holds, $\mathrm{t}e = \mathrm{t}\mathrm{d}\varphi = 0$ holds. That is, $e \in W^1(M, S_e)$.

Now, by requiring

$$\int_M \star\sigma \mathrm{d}\varphi \wedge \mathrm{d}\varphi' = \begin{cases} 0 & \text{when } \varphi' = \mathsf{n}^i \\ I_i & \text{when } \varphi' = \mathsf{N}^i \end{cases} \quad \forall\ \varphi' \in \{\mathsf{n}^i, \mathsf{N}^i\} \tag{4.20}$$

to hold, the j-side conditions in (4.5) and (4.6) are satisfied weakly, as well as the cohomology conditions (4.7) are satisfied similarly to the previous, more general case.

Actually, this strictly nodal method is also applicable when $\beta_1(M) \neq 0$ holds, but then the cohomology conditions that regard the elements of $H_1(M)$ are implicitly set to zero. That is, there can be no non-zero electromotive force around any tunnel of M. This is the shortcoming of the traditional choice of nodal basis functions only.

4.2.4 Current density -conforming formulation

In the current density conforming formulation, we seek to satisfy $\mathrm{d}j = 0$ and $\mathrm{t}_{S_j} j = 0$ exactly, and $\mathrm{d}e = 0$ and $\mathrm{t}_{S_e} e$ in the weak sense. That is, the sought solution is such that $[j] \in H^2_{\mathrm{dR}}(M, S_j)$ holds. The conditions $\mathrm{d}j = 0$ and $\mathrm{t}_{S_j} j = 0$ are satisfied by

$$j = \sum_{\sigma_i^1 \in M \setminus S_j} t_i \mathrm{d}\mathsf{e}^i + \sum_{i=1}^{\beta_2(M,S_j)} I_i \mathsf{F}^i \in W^2(M, S_j), \tag{4.21}$$

where the cohomology basis functions are of the form $\mathsf{F}^i \in HW^2(M, S_j)$, i.e. $[\mathsf{F}^i]$ span $\in H^2_{\mathrm{dR}}(M, S_j)$. Again, the support of the edge basis functions e do not cover S_j in order to leave the cohomology condition to be fixed by the cohomology basis functions.

We require that

$$\int_M \sigma^{-1} \star j \wedge j' = \begin{cases} 0 & \text{when } j' = \mathrm{d}\mathsf{e}^i \\ V_i & \text{when } j' = \mathsf{F}^i \end{cases} \quad \forall\ j' \in \{\mathrm{d}\mathsf{e}^i, \mathsf{F}^i\} \tag{4.22}$$

hold. Then, the e-side conditions $\mathrm{d}e = \mathrm{d}\sigma^{-1}\star j = 0$ and $\mathrm{t}_{S_e} e = \mathrm{t}\sigma^{-1}\star j = 0$ in (4.5) and (4.6) are satisfied weakly, since

$$0 = \int_M \sigma^{-1}\star\, j \wedge \mathrm{d}\mathsf{e}^i = -\int_M \mathrm{d}\sigma^{-1}\star\, j \wedge \mathsf{e}^i + \int_{S_e} \mathrm{t}\sigma^{-1}\star\, j \wedge \mathrm{t}\mathsf{e}^i \quad \forall\, \mathsf{e}^i \tag{4.23}$$

holds. By plugging (4.21) in (4.22) one obtains a linear system with blocks

$$\begin{bmatrix} \int_M \sigma^{-1}\star\mathsf{de}\wedge\mathsf{de} & \int_M \sigma^{-1}\star\mathsf{F}\wedge\mathsf{de} \\ \int_M \sigma^{-1}\star\mathsf{de}\wedge\mathsf{F} & \int_M \sigma^{-1}\star\mathsf{F}\wedge\mathsf{F} \end{bmatrix} \begin{bmatrix} \mathbf{t} \\ \mathbf{I} \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \mathbf{V} \end{bmatrix} \tag{4.24}$$

If all the coefficients I_i are fixed, this reduces to

$$\left[\int_M \sigma^{-1}\star\mathsf{de}\wedge\mathsf{de}\right]\left[\mathbf{t}\right] = -\left[\int_M \sigma^{-1}\star\mathsf{F}\wedge\mathsf{de}\right]\left[\mathbf{I}\right] \tag{4.25}$$

As a solution, one obtains such $j \in \mathcal{H}^2(M, S_j)$ that satisfies the current cohomology conditions in (4.7) for the chosen cohomology basis functions F^i. By fixing all the coefficients V_i instead, one obtains a solution that satisfies the voltage cohomology conditions in (4.7), since

$$V_i = \int_M \sigma^{-1}\star j \wedge \mathsf{F}^i = \int_M e \wedge \mathsf{F}^i = \int_{z_i^1} e \quad \forall\, 1 \le i \le \beta_1(M, S_e),\ [z_i^1] \in H_1(M, S_e), \tag{4.26}$$

holds by the equation (4.22) and by the properties of the cup product of cohomology.

Again in the general case, one can fix either V_i or I_i from each pair (V_i, I_i), but not both, to obtain an unique solution to the electrokinetic problem (4.5) - (4.7).

In the special case that $\beta_2(M) = 0$ holds, one can impose the cohomology conditions (4.7) from the boundary $S_j \subset \partial M$. This is analogous to the traditional electric field oriented formulation, where the cohomology conditions were imposed from to the boundary $S_e \subset \partial M$ using the cohomology basis functions of $H^0_{\mathrm{dR}}(S_e)$. Such basis functions can be obtained simply by summing up all the Whitney 0-forms associated with the 0-cells of each connected component of S_e

However, in the current density conforming formulation the situation is a bit more complicated, since the cohomology basis functions of $H^1_{\mathrm{dR}}(S_j)$ are not so simple to produce, but full scale cohomology computation on a 2-manifold is needed.

So, let $\beta_2(M) = 0$ and $[z^2] \in H_2(M, S_j)$ hold. By the long exact homology sequence, $0 \neq \partial_*[z^2] = [z^1] \in H_1(S_j)$ holds. An approximate solution of $j \in \mathcal{H}^k(M, S_j)$ can be expressed as an exterior derivative of a 1-form

$$t = \sum_{\sigma_i^1 \in M \setminus S_j} t_i \mathsf{e}^i + \sum_{i=1}^{\beta_1(S_j)-\beta_1(M)} I_i \mathsf{E}^i \in W^1(M), \tag{4.27}$$

where the cohomology basis functions $\mathsf{E}^i \in W^1(M)$ and satisfy $[\mathrm{t}\mathsf{E}^i] \in H^1_{\mathrm{dR}}(S_j)$. Again, not the full basis of $H^1_{\mathrm{dR}}(S_j)$ is used, bust just the ones that have non-zero image in $H^2_{\mathrm{dR}}(M, S_j)$. Since $\mathrm{t}_{S_j}\mathrm{d}\mathsf{E}^i = 0$ holds, $\mathrm{t}_{S_j} j = \mathrm{t}\mathrm{d}t = 0$ holds, i.e. $j \in W^2(M, S_j)$.

We require that

$$\int_M \sigma^{-1}\star\, \mathrm{d}t \wedge \mathrm{d}t' = \begin{cases} 0 & \text{when } t' = \mathsf{e}^i \\ V_i & \text{when } t' = \mathsf{E}^i \end{cases} \quad \forall\, t' \in \{\mathsf{e}^i, \mathsf{E}^i\} \tag{4.28}$$

hold. Then, the e-side conditions and in (4.5) and (4.6) are satisfied weakly, as well as the cohomology conditions (4.7) are satisfied similarly to the previous, more general case.

Again, this method is applicable when $\beta_2(M) \neq 0$ holds, but then the cohomology conditions regarding the elements of $H_2(M)$ are implicitly set to zero. That is, there cannot be a current source inside any void of M.

4.2.5 Electrostatic problem

The electrostatic problem generalizes the electrokinetic problem on two respects. First, there can be a source term, and second, a non-homogeneous boundary condition.

In the electrostatic problem, one is to find a pair of non-harmonic forms $(e, d) \in \Omega^1(M) \times \Omega^2(M)$ on the 3-manifold M that satisfy

$$\mathrm{d}e = 0, \qquad \mathrm{d}d = \rho, \qquad d = \epsilon \star e, \tag{4.29}$$

$$\mathrm{t}_{S_e} e = 0, \qquad \mathrm{t}_{S_d} d = \sigma \tag{4.30}$$

for the electric field e and the electric flux density d. The 3-form ρ stands for the charge density in M and the 2-form σ stands for the surface charge density on S_d. As usual, $S_e \subset \partial M$ and $S_d = \partial M \setminus S_e$ hold. If the problem is source-free and has homogeneous boundary conditions, i.e. $\mathrm{d}d = \rho = 0$ and $\mathrm{t}_{S_d} d = \sigma$ hold, one is back to the harmonic case and can exploit the cohomology basis functions exactly as in the electrokinetic problem.

In the case where $\mathrm{d}d = \rho \neq 0$ or $\mathrm{t}_{S_d} d = \sigma \neq 0$ holds, one can use an electric field conforming formulation with cohomology conditions. We impose the cohomology condition only on the closed and trace-free part of d. That is, we decompose $d = d_z + d_s$, where $\mathrm{d}d_z = 0$, $\mathrm{t}_{S_d} d_z = 0$, $\mathrm{d}d_s = \rho$, and $\mathrm{t}_{S_d} d_s = \sigma$ hold. The cohomology conditions are

$$\int_{z_i^1} e = V_i \quad \forall\, [z_i^1] \in H_1(M, S_e), \qquad \text{and} \qquad \int_{z_i^2} d_z = Q_i \quad \forall\, [z_i^2] \in H_2(M, S_d), \tag{4.31}$$

of which only $\beta_1(M, S_e) = \beta_2(M, S_d)$ are independent. The rest are determined as a part of the solution to the problem. These cohomology conditions can be interpreted as voltages on the terminals and net electric fluxes through the terminals.

We solve for an electric field $e = \mathrm{d}\varphi \in H\Omega^1(M, S_e)$ using a scalar potential $\varphi \in H\Omega^1(M)$:

$$\varphi = \sum_{\sigma_i^0 \in M \setminus S_e} \varphi_i \mathsf{n}^i + \sum_{i=1}^{\beta_0(S_e) - \beta_1(M)} V_i \mathsf{N}^i \in W^0(M), \tag{4.32}$$

where the cohomology basis functions $\mathsf{N}^i \in W^0(S_e)$ satisfy $[\mathsf{N}^i] \in H^0_{\mathrm{dR}}(S_e)$.

By requiring that

$$\int_M \star\epsilon \mathrm{d}\varphi \wedge \mathrm{d}\varphi' = \begin{cases} -\int_M \rho \wedge \mathsf{n}^i + \int_{S_d} \sigma \wedge \mathrm{t}\mathsf{n}^i & \text{when } \varphi' = \mathsf{n}^i \\ -\int_M \rho \wedge \mathsf{N}^i + \int_{S_d} \sigma \wedge \mathsf{N}^i + Q_i & \text{when } \varphi' = \mathsf{N}^i \end{cases} \quad \forall\, \varphi' \in \{\mathsf{n}^i, \mathsf{N}^i\} \tag{4.33}$$

hold, we get an approximate solution for the electrostatic problem (4.29) - (4.31).

4.2.6 Magnetostatic problem

The magnetostatic problem is analogous to the electrostatic problem. The only difference is that now the 1-form h may have source terms instead of the 2-form b. In comparison, in the electrostatic problem the situation was other way around, as the 2-form d might have had source terms, and the 1-form e was assumed source-free.

In the magnetostatic problem, one is to find a pair of non-harmonic forms $(h, b) \in \Omega^1(M) \times \Omega^2(M)$ on the 3-manifold M that satisfy

$$\mathrm{d}h = j, \qquad \mathrm{d}b = 0, \qquad b = \mu \star h, \tag{4.34}$$

$$\mathrm{t}_{S_h} h = k, \qquad \mathrm{t}_{S_b} b = 0 \tag{4.35}$$

for the magnetic field h and the magnetic flux density b. Here 2-form j is the current density in M and 1-form k is the surface current density on S_h. Again, $S_h \subset \partial M$ and $S_b = \partial M \setminus S_h$ hold. If the problem is source-free and has homogeneous boundary conditions, i.e. $\mathrm{d}h = j = 0$ and $\mathrm{t}_{S_h} h = 0$ hold, one is back to the harmonic case and one can exploit the cohomology basis functions exactly as in the electrokinetic problem.

Historically, an important special case of the source-free magnetostatic problem occurs when $S_b = \partial M$ hold and $\beta_1(M) = \beta_1(M, S_h) \geq 1$ hold. That is, the problem domain is magnetically isolated, and there are no sources of magnetic field inside the domain, yet a non-zero magnetic field can exist inside the domain. The study of such problem led to the awareness of homology and cohomology in the electromagnetic modeling, as it was important in applications. For example, in reluctance and inductance computations.

In the case where $\mathrm{d}h = j \neq 0$ or $\mathrm{t}_{S_h} h = k \neq 0$ hold, one can use a magnetic flux density conforming formulation with cohomology conditions. We impose the cohomology condition only on the closed part of h. That is, we decompose $h = h_z + h_s$, where $\mathrm{d}h_z = 0$, $\mathrm{d}h_s = j$, $\mathrm{t}_{S_h} h_z = 0$, and $\mathrm{t}_{S_h} h_s = k$ hold. The cohomology conditions are

$$\int_{z_i^2} b = M_i \quad \forall\, [z_i^2] \in H_2(M, S_b), \qquad \text{and} \qquad \int_{z_i^1} h_z = I_i \quad \forall\, [z_i^1] \in H_1(M, S_h), \tag{4.36}$$

of which only $\beta_2(M, S_b) = \beta_1(M, S_h)$ are independent. The rest are determined as a part of the solution to the problem. These cohomology conditions can be interpreted as net magnetic fluxes through surfaces and net currents through surfaces.

We solve for an magnetic field $b = \mathrm{d}a \in H\Omega^2(M, S_b)$ using a vector potential $a \in H\Omega^1(M)$:

$$a = \sum_{\sigma_i^1 \in M \setminus S_b} a_i \mathsf{e}^i + \sum_{i=1}^{\beta_1(S_b) - \beta_1(M)} M_i \mathsf{E}^i \in W^1(M), \tag{4.37}$$

where the cohomology basis functions $\mathsf{E}^i \in W^1(S_b)$ satisfy $[\mathsf{E}^i] \in H^1_{\mathrm{dR}}(S_b)$.

By requiring that

$$\int_M \mu^{-1} \star \mathrm{d}a \wedge \mathrm{d}a' = \begin{cases} \int_M j \wedge \mathsf{e}^i - \int_{S_h} k \wedge \mathrm{t}\mathsf{e}^i & \text{when } a' = \mathsf{e}^i \\ \int_M j \wedge \mathsf{E}^i - \int_{S_h} k \wedge \mathsf{E}^i + M_i & \text{when } a' = \mathsf{E}^i \end{cases} \quad \forall\, a' \in \{\mathsf{e}^i, \mathsf{E}^i\} \tag{4.38}$$

hold, we get an approximate solution for the magnetostatic problem (4.34) - (4.36).

4.3 Circuit coupled eddy current problem

We consider a special case of the eddy current problem, called **coupled field-circuit problem**, which can be naturally coupled to electric circuits. In addition to the magnetoquasistatic assumption, we assume that

1. On the domain boundary ∂M, electric field is closed. I.e. $\mathrm{t}_{\partial M}\mathrm{d}e = 0$ holds.
2. No magnetomotive nor electromotive forces are induced in the domain via magnetic fluxes through the holes of the domain boundary, i.e. i.e. $\int_{z^1} e = \int_{z^1} h = 0 \; \forall \; [z^1] \in H_1(\partial M)$ holds.
3. No magnetic flux or electric current emanates from the voids of M, i.e. $\int_{z^2} j = \int_{z^2} b = 0 \; \forall \; [z^2] \in H_2(\partial M)$ holds.

That is, we have forbidden the magnetic circuit coupling and the electromagnetic wave coupling of the problem. What is left is the electric circuit coupling.

What makes the formulations for the eddy current problem complex but also efficient to solve is that a part of the domain is assumed non-conducting. The different behavior of the fields in the conducting and in the non-conducting subdomains makes the problem two coupled problems on these two subdomains. We will show that the coupling between these two subdomains, as well as the coupling to the external circuit, is tightly related to the cohomology basis functions.

4.3.1 Eddy current problem

In the circuit coupled eddy current problem on a 3-manifold M we assume that M is divided in two parts: the conducting submanifold M_c and insulating submanifold M_a so that $M = M_c \cup M_a$ and $M_c \cap M_a = \partial M_c \cap \partial M_a$ hold. We look for a pairs of differential forms $(e, j) \in \Omega^1(M_c) \times \Omega^2(M_c)$ and $(h, b) \in \Omega^1(M) \times \Omega^2(M)$ on a 3-manifold M that satisfy

$$\mathrm{d}e = -\partial_t b \qquad \mathrm{d}j = 0 \qquad j = \star\sigma e \tag{4.39}$$

$$\mathrm{t}_{S_e} e = 0 \qquad \mathrm{t}_{S_j} j = 0 \tag{4.40}$$

$$\mathrm{d}h = j \qquad \mathrm{d}b = 0 \qquad b = \star\mu h \tag{4.41}$$

$$\mathrm{t}_{\partial M} b = 0, \tag{4.42}$$

where $\partial M_c = S_e \cup S_j$ so that $\partial S_e = \partial S_j = S_e \cap S_j$ hold. In addition, require that the following hold:

$$\mathrm{d}h = 0 \text{ on } M_a \qquad \text{and} \qquad \mathrm{t}j = 0 \text{ on } \partial M_c \cap \partial M_a \subset S_j. \tag{4.43}$$

That is, the insulating submanifold has no currents and the current cannot pass through the interface $\partial M_c \cap \partial M_a$ between the insulating submanifold M_a and the conducting submanifold M_c.

The field $\mathrm{t}h$ is closed on $S_j \cap \partial M$, since $\mathrm{d}\mathrm{t}h = \mathrm{t}\mathrm{d}h = 0$ holds on S_j. Thus, we may assign meaningful cohomology conditions

$$V_i = \int_{z_i^1} \mathrm{t}e \quad [z_i^1] \in H_1(\partial M \setminus S_e, \partial S_e), \qquad \text{and} \qquad I_i = \int_{z_i^1} \mathrm{t}h \quad [z_i^1] \in H_1(\partial M \setminus S_e). \tag{4.44}$$

They correspond to voltage differences V_i between the circuit terminals on S_e and net currents I_i from terminal to terminal on S_e. If S_e has N connected components, we have $N-1$ independent cohomology conditions.

In the so-called Faraday's law and Ampere's law conforming formulations these cohomology conditions can be related to the harmonic part of the electric field in M_c and to the harmonic magnetic field in M_a. That is, they related to the basis elements of the cohomology spaces $H^1(M_c, S_e)$ and $H^1(M_a)$.

The method to exploit cohomology basis functions in the formulations builds on the ideas developed in the electrokinetic problem. However, since the magnetic fluxes and the electric currents emanating from the voids of the domain were assumed not to exist, only cohomology edge basis functions are needed.

4.3.2 Faraday's law conforming formulation

In the Faraday's law conforming formulation we solve for the harmonic part $\varepsilon \in \mathcal{H}^1(M_c, S_e)$ of the electric field e and for the magnetic vector potential $a \in H\Omega^1(M, \partial M)$. Then, the eddy currents j in M_c are $j = \star\sigma e = \star\sigma(\varepsilon - \partial_t a)$.

The magnetic vector potential a is approximated as

$$a = \sum_{\sigma_i^1 \in M \setminus \partial M} a_i \mathsf{e}^i \in W^1(M, \partial M) \tag{4.45}$$

and the harmonic part of the electric field is approximated as

$$\varepsilon = -\sum_{\sigma_i^0 \in M_c \setminus S_e} \varphi_i \mathrm{d}\mathsf{n}^i + \sum_{i=1}^{\beta_1(M_c, S_e)} V_i \mathsf{E}^i \in W^1(M_c, S_e), \tag{4.46}$$

where the coefficients V_i determine the cohomology class of $[\varepsilon] \in H^1_{\mathrm{dR}}(M, S_e)$. These approximations satisfy the boundary conditions $\mathrm{t}_{S_e} e = \mathrm{t}_{\partial M} b = 0$ and the Faraday's and Gauss law for $e = \varepsilon - \partial_t a$ and $b = \mathrm{d}a$. The rest of the boundary conditions and partial differential equations are to be satisfied weakly.

We require

$$\int_M \mu^{-1} \star \mathrm{d}a \wedge \mathrm{d}a' + \int_{M_c} \sigma \star \varepsilon \wedge a' - \int_{M_c} \star\sigma \partial_t a \wedge a' = 0 \quad \forall\, a' \in \{\mathsf{e}^i\} \tag{4.47}$$

$$\int_{M_c} \sigma \star \varepsilon \wedge \varepsilon' - \int_{M_c} \sigma \star \partial_t a \wedge \varepsilon' = \begin{cases} 0 \text{ when } \varepsilon' = \mathrm{d}\mathsf{n}^i \\ I_i \text{ when } \varepsilon' = \mathsf{E}^i \end{cases} \qquad \forall\, \varepsilon' \in \{\mathrm{d}\mathsf{n}^i, \mathsf{E}^i\} \tag{4.48}$$

The equation (4.47) enforces Ampere's law $\mathrm{d}h = j$ weakly. The equation (4.48) enforces $\mathrm{d}j = 0$ and $\mathrm{t}_{S_j} j = 0$ weakly, similarly as in the electric field conforming formulation of the electrokinetic problem.

The cohomology class of $[\varepsilon]$ is related to the cohomology conditions of the eddy current problem. The coefficients V_i of the cohomology basis functions E^i correspond to a voltage difference when circulating around a tunnel in M_c and between terminals S_e connected by M_c. By the equation (4.48), they also relate to the net currents I_i around tunnels in M_c and from terminal to terminal on S_e.

By plugging (4.45) and (4.46) into equations (4.47) and (4.48), one obtains a linear system

$$\mathrm{M}\begin{bmatrix}\dot{\boldsymbol{\varphi}}\\ \dot{\mathbf{a}}\\ \dot{\mathbf{V}}\end{bmatrix} + \mathrm{A}\begin{bmatrix}\boldsymbol{\varphi}\\ \mathbf{a}\\ \mathbf{V}\end{bmatrix} = \begin{bmatrix}\mathbf{0}\\ \mathbf{0}\\ \mathbf{I}\end{bmatrix}, \tag{4.49}$$

where the matrices M and A have blocks

$$\mathrm{M} = \begin{bmatrix} 0 & \int_{M_c} \star\sigma \mathsf{e} \wedge \mathrm{d}\mathsf{n} & 0 \\ 0 & \int_{M_c} \star\sigma \mathsf{e} \wedge \mathsf{e} & 0 \\ 0 & \int_{M_c} \star\sigma \mathsf{e} \wedge \mathsf{E} & 0 \end{bmatrix} \text{ and} \tag{4.50}$$

$$\mathrm{A} = \begin{bmatrix} \int_{M_c} \star\sigma \mathrm{d}\mathsf{n} \wedge \mathrm{d}\mathsf{n} & 0 & -\int_{M_c} \star\sigma \mathsf{E} \wedge \mathrm{d}\mathsf{n} \\ \int_{M_c} \star\sigma \mathrm{d}\mathsf{n} \wedge \mathsf{e} & \int_M \mu^{-1} \star \mathrm{d}\mathsf{e} \wedge \mathrm{d}\mathsf{e} & -\int_{M_c} \star\sigma \mathsf{E} \wedge \mathsf{e} \\ \int_{M_c} \star\sigma \mathrm{d}\mathsf{n} \wedge \mathsf{E} & 0 & -\int_{M_c} \star\sigma \mathsf{E} \wedge \mathsf{E} \end{bmatrix}. \tag{4.51}$$

4.3.3 Ampere's law conforming formulation

In the Ampere's law conforming formulation we solve for the magnetic field $h \in H\Omega^1(M)$. Then, the eddy currents j in M_c are $j = \mathrm{d}h$. Since $\mathrm{d}h = 0$ holds in M_a, the magnetic field is harmonic in M_a, i.e. the restriction $h \in \mathcal{H}^1(M_a, S_h)$.

The magnetic field h is approximated as

$$h = \sum_{\sigma_i^0 \in M_a} \phi_i \mathrm{d}\mathsf{n}^i + \sum_{\sigma_i^1 \in M_c \setminus S_j} t_i \mathsf{e}^i + \sum_{i=1}^{\beta_1(M_a)} I_i \mathsf{E}^i \in W^1(M) \subset W^1(M, S_j), \tag{4.52}$$

where the coefficients I_i determine the cohomology class of $[h] \in H^1_{\mathrm{dR}}(M_a)$. This approximation for the magnetic field satisfies the boundary condition $\mathrm{t}j = 0$ on S_j, and the Ampere's law $\mathrm{d}h = j$ ($= 0$ in M_a), $\mathrm{d}j = 0$. The rest of the boundary conditions and partial differential equations are to be satisfied weakly.

We require

$$\int_M \star\mu h \wedge h' = 0 \quad \forall\, h' \in \{\mathrm{d}\mathsf{n}^i, \mathsf{e}^i, \mathsf{E}^i\} \tag{4.53}$$

$$\int_{M_c} \sigma^{-1} \star \mathrm{d}h \wedge \mathrm{d}h' + \int_M \partial_t \star \mu h \wedge h' = \begin{cases} 0 & \text{when } h' = \mathrm{d}\mathsf{n}^i \text{ or } h' = \mathsf{e}^i \\ V_i & \text{when } h' = \mathsf{E}^i \end{cases} \quad \forall\, h' \in \{\mathrm{d}\mathsf{n}^i, \mathsf{e}^i, \mathsf{E}^i\}. \tag{4.54}$$

The equation (4.53) enforces $\mathrm{d}b = 0$ and $\mathrm{t}_{\partial M} b = 0$ weakly. The equation (4.54) enforces the Faraday's law $\mathrm{d}e = -\partial_t b$ and the boundary condition $\mathrm{t}_{S_e} e = 0$ weakly. This is similar how the current density conforming formulation of the electrokinetic problem enforced $\mathrm{d}e = \mathrm{t}_{S_e} e = 0$ weakly.

The cohomology class of $[h]$ is related to the cohomology conditions of the eddy current problem. The coefficients I_i of the cohomology basis functions E^i correspond to currents through tunnels in M_a. By the equation (4.54), they also relate to the electromotive forces V_i around the tunnels in M_a and voltage differences between electric terminals on S_e.

By plugging (4.52) in the equations (4.53) and (4.54) one obtains a linear system of ordinary differential equations

$$\mathrm{M}\begin{bmatrix}\dot{\boldsymbol{\phi}}\\ \dot{\mathbf{t}}\\ \dot{\mathbf{I}}\end{bmatrix}+\mathrm{A}\begin{bmatrix}\boldsymbol{\phi}\\ \mathbf{t}\\ \mathbf{I}\end{bmatrix}=\begin{bmatrix}\mathbf{0}\\ \mathbf{0}\\ \mathbf{V}\end{bmatrix}, \tag{4.55}$$

where the matrices M and A have blocks

$$\mathrm{M}=\begin{bmatrix}\int_M \star\mu\mathsf{dn}\wedge\mathsf{dn} & \int_M \star\mu\mathsf{e}\wedge\mathsf{dn} & \int_M \star\mu\mathsf{E}\wedge\mathsf{dn}\\ \int_M \star\mu\mathsf{dn}\wedge\mathsf{e} & \int_M \star\mu\mathsf{e}\wedge\mathsf{e} & \int_M \star\mu\mathsf{E}\wedge\mathsf{e}\\ \int_M \star\mu\mathsf{dn}\wedge\mathsf{E} & \int_M \star\mu\mathsf{e}\wedge\mathsf{E} & \int_M \star\mu\mathsf{E}\wedge\mathsf{E}\end{bmatrix}, \text{ and} \tag{4.56}$$

$$\mathrm{A}=\begin{bmatrix}\int_M \star\mu\mathsf{dn}\wedge\mathsf{dn} & \int_M \star\mu\mathsf{e}\wedge\mathsf{dn} & \int_M \star\mu\mathsf{E}\wedge\mathsf{dn}\\ \int_M \star\mu\mathsf{dn}\wedge\mathsf{e} & \int_{M_c}\sigma^{-1}\star\mathsf{de}\wedge\mathsf{de}+\int_M \star\mu\mathsf{e}\wedge\mathsf{e} & \int_M \star\mu\mathsf{E}\wedge\mathsf{e}\\ \int_M \star\mu\mathsf{dn}\wedge\mathsf{E} & \int_M \star\mu\mathsf{e}\wedge\mathsf{E} & \int_M \star\mu\mathsf{E}\wedge\mathsf{E}\end{bmatrix}. \tag{4.57}$$

4.4 Examples

4.4.1 Induced EMF in squirrel cage rotor

In this example, we model the current density induced to the squirrel cage rotor of the induction motor as an electrokinetic problem. That is, we assume that there's no skin effect in the squirrel cage and that the external magnetic field is unaffected by the rotor currents. We model only a sector of the squirrel cage, see Fig. 4.4.

Such heavy assumptions are used here for the sake of clarity to showcase an isolated electrokinetic problem. In actual motor modeling, this problem would be coupled to the magnetic problem around the squirrel cage.

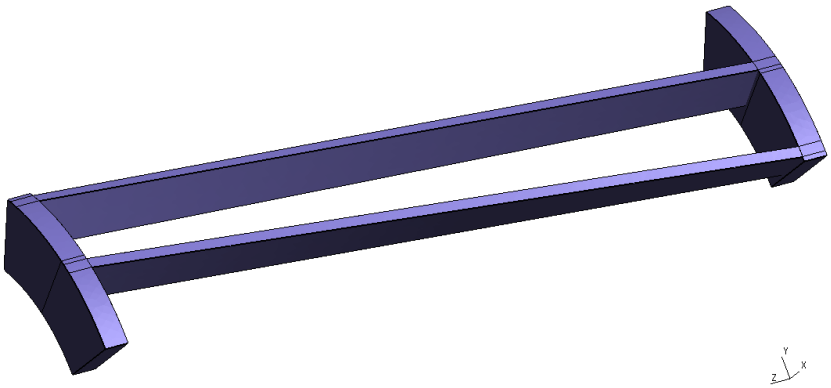


Figure 4.4: A sector of a squirrel cage in the rotor of an induction motor.

The modeling domain M is the copper-made squirrel cage in the rotor. As we model the current distribution as an electrokinetic problem, $\mathrm{d}e=\mathrm{d}j=0$ and $j=\star\sigma e$ hold in M. At the symmetry planes $\mathrm{t}e=0$ holds while $\mathrm{t}j=0$ holds elsewhere on the boundary ∂M. We denote the subdomains by S_e and S_j accordingly. The problem is driven by

the induced electromotive force around the holes in the squirrel cage. That is, by the cohomology conditions

$$V_i = \int_{z^1} e, \quad [z_i^1] \in H_1(M, S_e) \tag{4.58}$$

The dimension $\beta_1(M, S_e)$ of the vector space $H_1(M, S_e)$ is four. A basis can be found where the basis elements correspond to the three holes between the bars, while the last one corresponds to the end which connects the bars. Such basis and its cohomology cobasis is computed and depicted in Fig. 4.5.

To obtain such basis for $H_1(M, S_e)$, we have first computed a basis $H_1(M, S_e)$ and manually constructed a unimodular transformation to obtain the basis in Fig. 4.5. Then, we have computed its cohomology cobasis from a computed basis of $H^1(M, S_e)$ as described in 3.5.2. The elements of the cohomology basis are used to construct the cohomology basis functions for the problem. To solve the problem, we have set the coefficients of the cohomology basis functions as follows: $V_1 = V_2 = 0$ and $V_3 = V_4 = 1$.

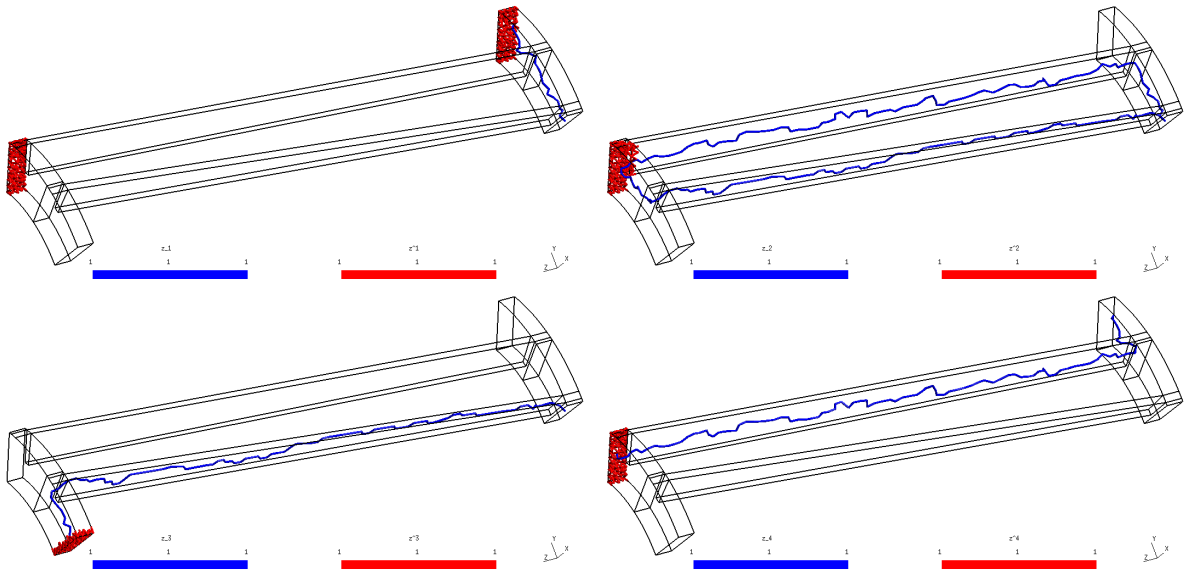


Figure 4.5: Homology and cohomology space bases for the squirrel cage. Each figure represents a representative of a basis element of $H_1(M, S_e)$ together with a representative of its cobasis element in $H^1(M, S_e)$. The cobasis elements of $H^1(M, S_e)$ are used to construct the cohomology basis functions for the problem.

Alternatively, one can use the current oriented formulation. To satisfy the cohomology conditions in (4.58), we have to compute a basis for $H^2(M, S_j)$ from which the cohomology basis functions are constructed. A basis that corresponds the basis of $H_1(M, S_e)$ in the electric field conforming formulation is presented in the Fig. 4.6.

4.4.2 Mutual inductance magnetostatic problem

In this example, we model air core coil together with a wire printed on a circuit board. The goal is to compute their mutual inductance as a magnetostatic problem with cohomology

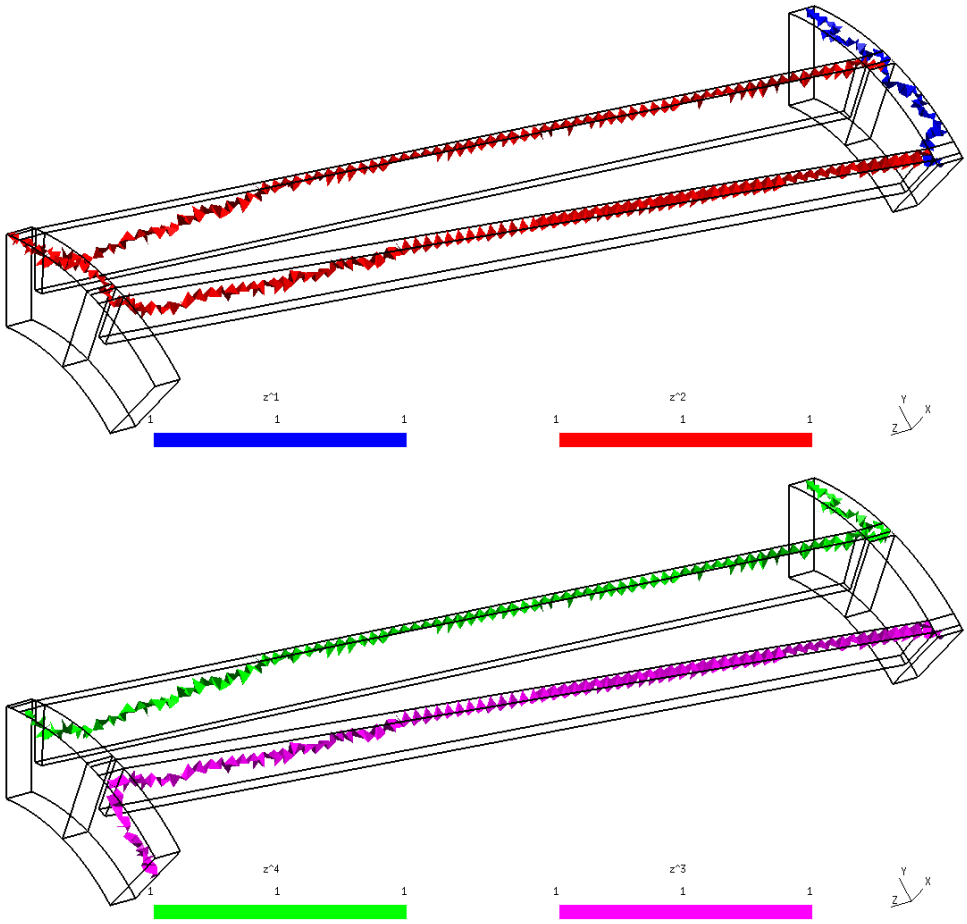


Figure 4.6: Representatives of the four basis elements of the cohomology space $H^2(M, S_j)$.

basis functions. Such a computation can be applied for radio frequency transformer design or to compute the electromagnetic interference of a coil and a wire on a crammed circuit board. We consider both magnetic field and magnetic flux density conforming formulations.

The motive of this example is to present how the coefficients of the cohomology basis functions can be used to compute lumped parameters, such as inductances in this example. Also, cohomology basis functions enable one to easily use both magnetic flux density and magnetic field intensity oriented formulations. Therefore, one can obtain complementary error bounds for the inductances, as argued in [8].

The modeling domain M is the non-magnetic, non-conducting region around the conductors, depicted in Fig. 4.7. That is, on M $\mathrm{d}h = 0$ holds. We assume the magnetic flux does not penetrate the domain boundary ∂M, i.e. $\mathrm{t}b = 0$ holds there. Such is a good approximation on high frequencies when the conductor skin depth is negligible.

The cohomology conditions of the problem are either the net currents I_i in the coil and in the wire or the net magnetic fluxes M_i caused by the currents. At high frequencies when $\mathrm{t}b = 0$ approximately holds, their ratios approximate mutual and self-inductances

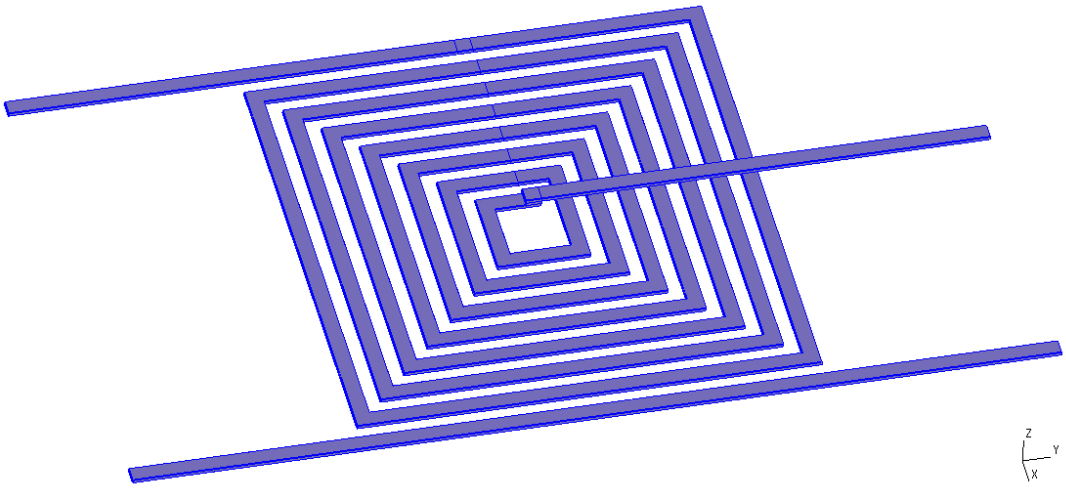


Figure 4.7: The modeling domain of the mutual inductance magnetostatic problem is the non-magnetic, non-conducting region outside the conductors.

of the coil and the wire.

$$I_i = \int_{z^1} h, \quad [z_i^1] \in H_1(M), \qquad \text{and} \qquad M_s = \int_{z_i^2} b, \quad [z_i^2] \in H_2(M, \partial M). \tag{4.59}$$

In the magnetic field conforming formulation the cohomology conditions are associated to the cohomology basis functions $\mathsf{E}^i \in HW^1(M)$. For their easy interpretation in the formulation, their coefficients should equal to the current in the coil and in the wire. Let the homology space $H_1(M)$ basis be such that the representatives z_i^1 of the basis elements loop only around either the coil or the wire once. Then, the coefficients of the cohomology basis functions E^i constructed from the cobasis of $H^1(M)$ correspond to the net current I_i in the coil or in the wire.

There are three strategies to obtain such cohomology basis functions, as noted in the section 4.2.2 on page 84. Let M_{c1} and M_{c2} denote coil and wire submanifolds and S_j the insulating boundary around them.

1. Compute basis $\{[z^1]\}$ for $H^1(M \cup M_{c1})$ and basis $\{[z^2]\}$ for $H^1(M \cup M_{c2})$ separately. By the Mayer-Vietoris sequence, $\{[\tilde{i}z^2], [\tilde{i}z^2]\}$ spans $H^1(M)$, where $\tilde{i} : C^1(M \cup M_{c1} \cup M_{c2}) \to C^1(M)$.

2. Compute bases for $H_2(M_{c1}, S_j)$, $H_2(M_{c2}, S_j)$, and $H^1(M)$. Then the boundaries of the basis elements of $H_2(M_{c1}, S_j)$, $H_2(M_{c2}, S_j)$ span $H_1(M)$ by the long exact homology sequence. Then transform the computed basis of $H^1(M)$ to be the cobasis of $H_1(M)$.

3. Compute $H_1(M)$, $H^1(M)$. Then manually transform the basis $H_1(M)$ to be such that the representatives loop once around one coil only. Then transform the computed basis of $H^1(M)$ to be the cobasis of $H_1(M)$.

In Fig. 4.8 are the edges of M that have non-zero coefficients in the cohomology basis functions E^1 and E^2 in $HW^1(M)$.

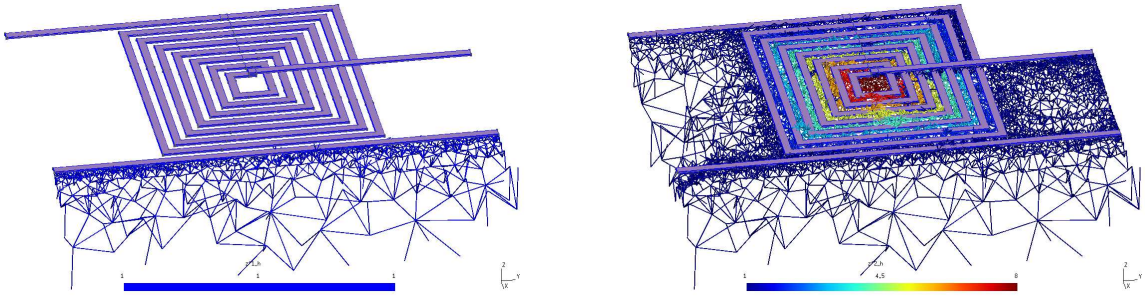


Figure 4.8: Edges of M that have non-zero coefficients in the cohomology basis functions $\mathsf{E}^1, \mathsf{E}^2 \in HW^1(M)$

In the magnetic flux density conforming formulation the cohomology conditions are associated to the cohomology basis functions $\mathsf{F}^i \in HW^2(M, \partial M)$. To have them correspond to the currents in the coil and in the wire, they should be constructed from a basis of $H^2(M, \partial M)$ where the "dual edges" of the faces of the representatives loop once around the coil or the wire. Such basis is depicted in Fig. 4.9.

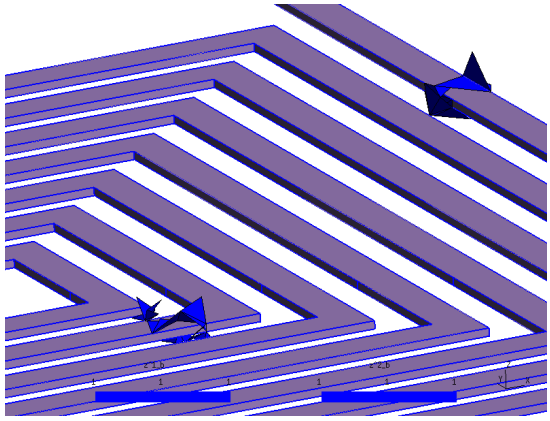


Figure 4.9: Faces of M that have non-zero coefficients in the cohomology basis functions $\mathsf{F}^1, \mathsf{F}^2 \in HW^2(M, \partial M)$

In Table 4.1 we have compared the magnetic field and magnetic flux density conforming formulations of the magnetostatic problem. The number of degrees of freedom is tremendously larger in the magnetic flux density conforming formulation. Also note that the complementarity [8] is visible from the results: the magnetic flux density conforming formulation approximates the inductances from below, and the magnetic field conforming one from above. In Fig. 4.10 are cross-sections of the computed magnetic flux densities using the magnetic field and magnetic flux density conforming formulations, respectively.

4.4.3 Induction heating eddy current problem

In this example, we model an induction heating devise as an coupled field-circuit eddy current problem. The problem is solved using the formulations of the section 4.3 on page

Table 4.1: Comparison of the magnetic flux density b and magnetic field intensity h conforming formulations for the magnetostatic problem. The mesh was generated using the Frontal [58] algorithm available in *Gmsh*. The linear system was solved using a direct solver MUMPS [1].

	b-conf.	h-conf. #1	h-conf. #2
Mesh tetrahedra	172 846	172 846	585 442
Mesh time [s]	8.5	8.5	34
Cohomology time [s]	18	34	160
Degrees of freedom	195 082	28 311	101 834
Solution time [s]	268	20	131
Inductance matrix [nH]	$\begin{bmatrix} 19.4 & 11.0 \\ 11.0 & 344 \end{bmatrix}$	$\begin{bmatrix} 26.1 & 11.4 \\ 11.4 & 417 \end{bmatrix}$	$\begin{bmatrix} 22.8 & 11.3 \\ 11.3 & 385 \end{bmatrix}$

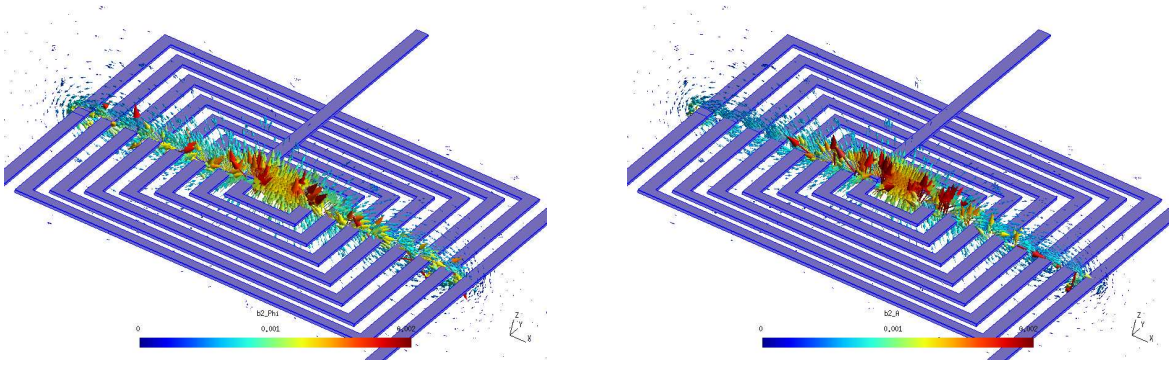


Figure 4.10: Cross-section of the magnetic flux density b in M computed using the magnetic field (left) and magnetic flux density (right) conforming formulations.

90. Although the geometry in this example is specifically tailored for a typical induction heating configuration, the example represents a large class of eddy current problems.

The modeling domain M is depicted in Fig. 4.11, which also reviews the local conditions assigned on the fields on different parts of the domain. In contrast to the local conditions, we also assign a cohomology condition on fields. The cohomology condition is either

$$V_s = \int_{z^1} \mathrm{t}e \quad [z^1] \in H_1(\partial M \setminus S_e, \partial S_e), \quad \text{or} \quad I_s = \int_{z^2} j = \int_{\partial z^2} \mathrm{t}h \quad [\partial z^2] \in H_1(\partial M \setminus S_e). \tag{4.60}$$

That is, either the source voltage V_s or source current I_s driven to the inductor coil, or their affine relation can be used to drive the problem. Since in this problem $H_1(M) = H_2(M) = 0$ holds, one infers from the long exact homology sequence that homology spaces $H_1(S_b \setminus S_e)$ and $H_2(M, \partial M \setminus S_e)$ are isomorphic. Therefore, if $H_2(M, \partial M \setminus S_e)$ is spanned by $[z^2]$, then $H_1(\partial M \setminus S_e)$ is spanned by $[\partial z^2]$. Such cycles z^1, z^2, and ∂z^2 are depicted in Fig. 4.12.

In Fig. 4.13 the model of the 3-manifold M is represented together with the surface mesh of M_c. If one wanted to use an Ampere's law conforming formulation without

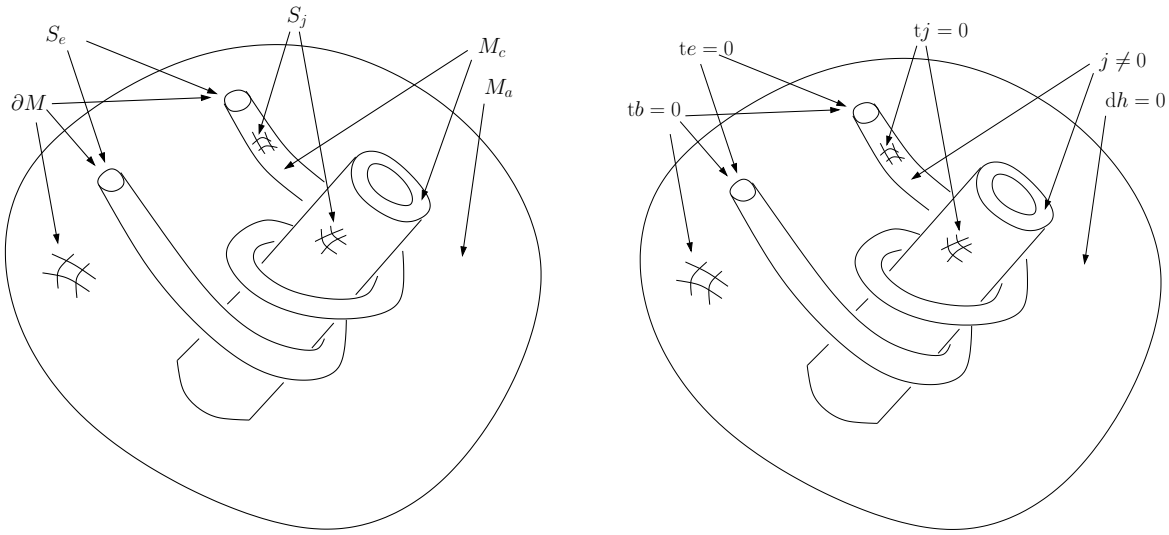


Figure 4.11: Induction heating eddy current problem domain and assigned local conditions on fields.

cohomology computation for such geometry, it would be quite difficult for a human or non-cohomology based algorithm to construct suitable cuts for the magnetic scalar potential.

In the Ampere's law conforming formulation of the eddy current problem, the cohomology conditions are related to the cohomology basis functions $\mathsf{E}^i \in HW^1(M_a)$. Since there are two tunnels through the submanifold M_a, $\beta_2(M_a) = 2$ holds. One can find a basis $\{[z_1^1], [z_2^1]\}$ of $H_1(M_a)$ such that $[z_1^1] = [\partial z^2]$ holds, where $[\partial z^2]$ spans $H_1(\partial M \setminus S_e)$. For such a basis, one can compute cohomology cobasis of $H^1(M_a)$, and construct a set $\{\mathsf{E}^1, \mathsf{E}^2\}$ of cohomology basis functions associated to the cobasis of $H^1(M_a)$. Then, the coefficient of the cohomology basis function E^1 will equal to the source current I_s in the problem formulation. Correspondingly, the coefficient V_1 in equation (4.54) will equal to the source voltage V_s.

One can use multiple strategies to find such set of cohomology basis functions, as noted in the section 4.2.2. Here, they are in inverse order of interference required from the user. However, the most automatic strategy is computationally the most expensive.

1. Compute bases for $H^1(M_a \cup M_{c1})$ and $H^1(M_a \cup M_{c2})$ separately. By the Mayer-Vietoris sequence, they span $H^1(M_a)$.

2. Compute bases for $H_2(M_{c1}, S_j)$, $H_2(M_{c2}, S_j)$, and $H^1(M_a)$. Then, the boundaries of basis elements of $H_2(M_{c1}, S_j)$, $H_2(M_{c2}, S_j)$ span $H_1(M_a)$ by the long exact homology sequence. Then transform the computed basis of $H^1(M_a)$ to be the cobasis of $H_1(M_a)$.

3. Compute bases for $H_1(M_a)$ and $H^1(M_a)$. Then manually transform the basis $H_1(M_a)$ to be such that other representative loops once around the inductor coil only. Then transform the computed basis of $H^1(M_a)$ to be the cobasis of $H_1(M_a)$.

In Fig. 4.14 are the edges of M_a that have non-zero coefficients in the cohomology basis functions E^1 and E^2 in $HW^1(M_a)$.

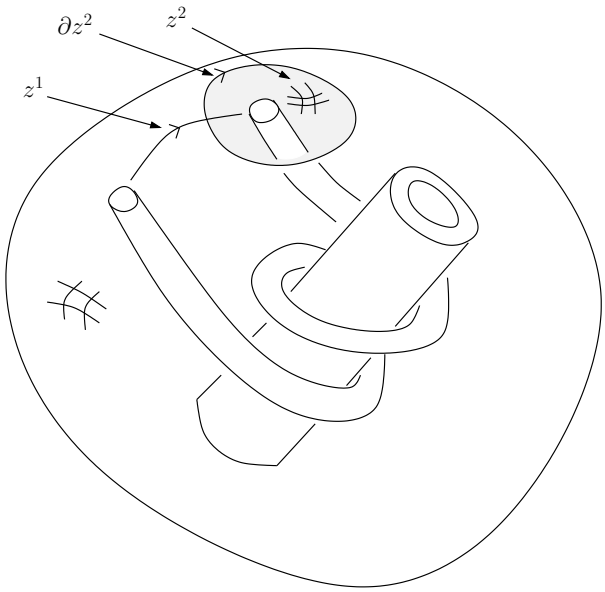


Figure 4.12: 1-cycles z^1 and ∂z^2, and a 2-cycle z^2 whose cosets span the homology spaces $H_1(S_b \setminus S_e, \partial S_e)$, $H_1(S_b \setminus S_e, \partial S_e)$, and $H_2(M, S_b \setminus S_e, \partial S_e)$, respectively. They appear in the cohomology conditions of the problem.

In the Faraday's law conforming formulation of the eddy current problem, the cohomology conditions are related to the cohomology basis functions $\mathsf{E}^i \in HW^1(M_c, S_e)$. Again, there are two tunnels modulo S_e through the submanifold M_c, i.e. $\beta_1(M_c, S_e) = 2$ holds. One can find a basis $\{[z_1^1], [z_2^1]\}$ of $H_1(M_c, S_e)$ such that $0 \neq [\partial z_1^1] \in H_0(S_e)$ holds. For such a basis, one can compute cohomology cobasis of $H^1(M_c, S_e)$, and construct a set $\{\mathsf{E}^1, \mathsf{E}^2\}$ of cohomology basis functions associated to the cobasis of $H^1(M_c, S_e)$. Then, the coefficient of the cohomology basis function E^1 will equal to the source voltage V_s in the problem formulation. Correspondingly, the coefficient I_1 in equation (4.48) will equal to the source current I_s. In this case, since M_c has two connected components, finding such cohomology basis functions using cohomology computation is easier.

In Fig. 4.15 are the edges of M_c that have non-zero coefficients in the cohomology basis functions E^1 and E^2 in $HW^1(M_c, S_e)$.

In the Table 4.2 we have compared the Ampere's law and Faraday's' law conforming formulation of the induction heating problem. Clearly, the number of degrees of freedom is tremendously larger in the Faraday's law conforming formulation. However, the cohomology computation time in the Ampere's law conforming formulation is longer. In figures 4.16 and 4.17 are the computed current densities in the conducting submanifold M_c using the Ampere's law and Faraday's law conforming formulations, respectively.

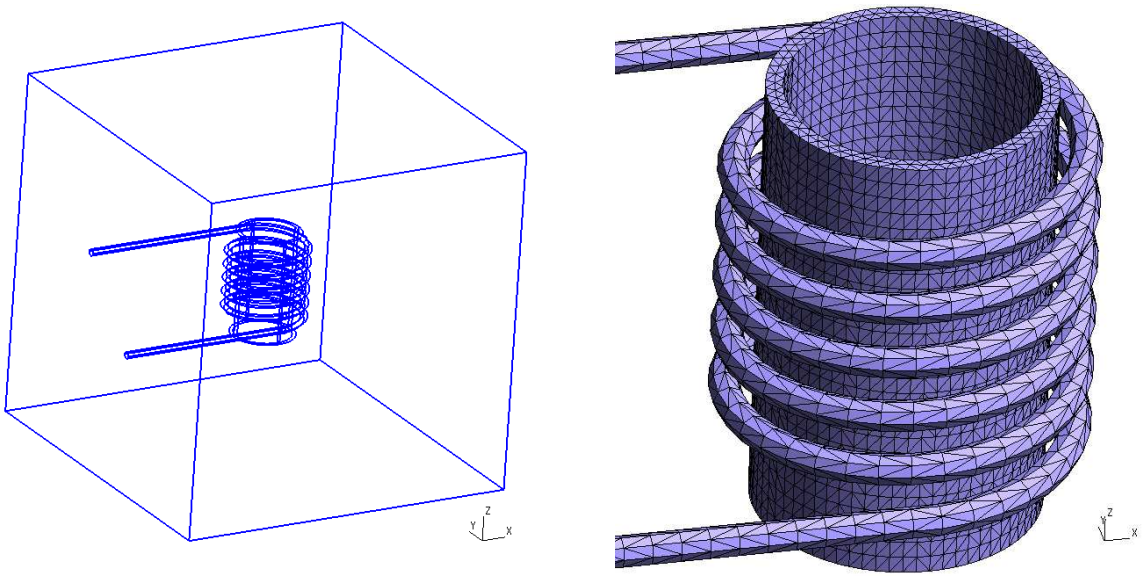


Figure 4.13: Induction heating problem model of M and surface mesh of M_c

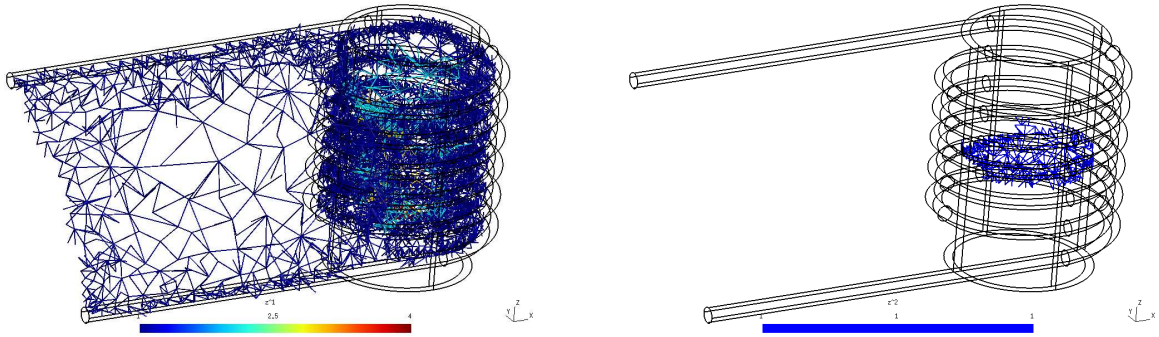


Figure 4.14: Edges of M_a that have non-zero coefficients in the cohomology basis functions $\mathsf{E}^1, \mathsf{E}^2 \in HW^1(M_a)$

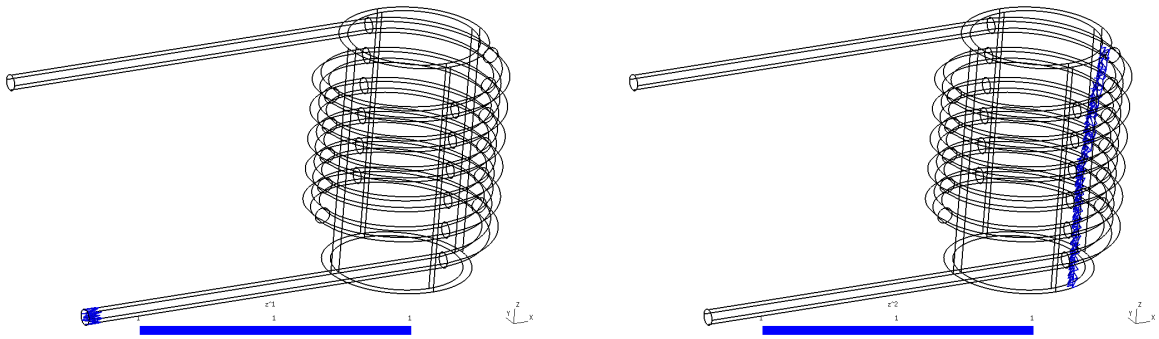


Figure 4.15: Edges of M_c that have non-zero coefficients in the cohomology basis functions $\mathsf{E}^1, \mathsf{E}^2 \in HW^1(M_c, S_e)$

Table 4.2: Comparison of the formulations for the induction heating eddy current problem. The mesh was generated using the Frontal 3D [58] algorithm available in *Gmsh*. The linear system was solved using a direct solver MUMPS [1]. The computation times presented here are wall-clock times. That is, they are here for comparison and to give a sense of the order of magnitude of the computation times.

	Faraday's law conf.	Ampere's law conf. #1	Ampere's law conf. #2
Mesh tetrahedra	113 502	113 502	280 417
Mesh time [s]	21	21	115
Cohomology time [s]	1	16	57
Degrees of Freedom	275 118	89 946	261 228
Solution time [s]	3 307	308	1 670
Impedance [mΩ]	$0.500 + i1.00$	$0.538 + i0.807$	$0.490 + i0.941$

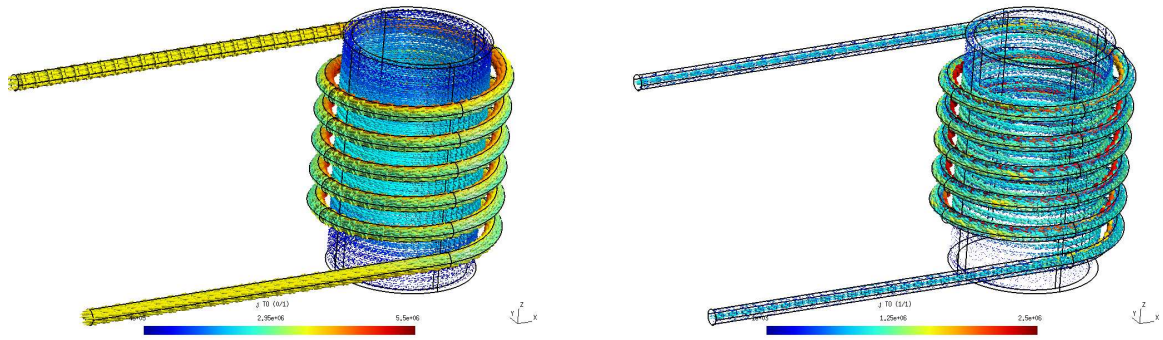


Figure 4.16: Current density j in M_c computed using the Ampere's law conforming formulation. The real part is on the left and the imaginary part is on the right.

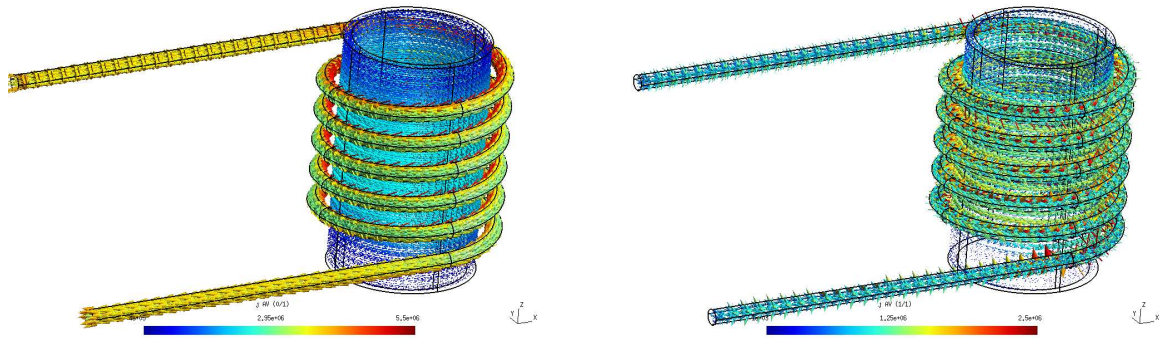


Figure 4.17: Current density j in M_c computed using the Faraday's law conforming formulation. The real part is on the left and the imaginary part is on the right.

Chapter 5

Finite element imitation of the Riemannian manifold

In this chapter, we describe a `C++` library that is used to imitate the structure and objects defined on the Riemannian manifold. The library is build on the finite element environment *Gmsh* [26], whose coordinate description of a finite element model provides a stepping stone for the Riemannian manifold abstraction.

The imitation emphasizes the separation of the manifold metric and coordinate numbers. That is, used the coordinate chart necessarily doesn't have a direct relation to the distances between the points on the manifold, and so-called *covariant* and *contravariant* tensors are strictly different kind objects.

The described finite element imitation of the Riemannian manifold is designed to help its user to program finite element pre- and post-processors for boundary value problems whose domain is a Riemannian manifold, rather than the Euclidean space, which is assumed in the traditional software systems.

This chapter is heavily based on the author's recent publication [52].

5.1 Motivation

The finite element method is traditionally considered to solve boundary value problems on subsets of the Euclidean space. Usually, the dimension and the coordinate system is fixed in advance or a few alternatives are provided, since the form of the partial differential equations in the classical vector analysis depend on them.

However, differential forms on manifolds offer an alternative approach, where the form of the partial differential equations no longer depend on the used coordinate system and the dimension. Further, one is not restricted to the Euclidean metric, but any metric tensor can be used without affecting the formulation of the boundary value problem.

The actual numerical computations still need to take the dimension, coordinate system, and the metric tensor into account. However, this can be done in a systematic manner, where they are defined in one place, and then automatically taken into account when needed. This is in contrast to the traditional approach, where one should take care at each turn when formulating a problem.

This Riemannian manifold imitation provides an interface where the slots for the dimension, coordinate system, and the metric tensor are offered. Further, it provides

numerical objects whose computational operations take the choices automatically into account.

5.2 Imitation of the manifold

In a finite element preprocessor, such as *Gmsh*, the domain of a boundary value problem is represented as *compact* subsets, or regions, of $\mathbb{R}^3$ that constitute a geometric model. A finite element mesh generator produces a finite element mesh consisting of finite elements of dimension $m \leq 3$ that cover the m-regions of corresponding dimension.

In the Riemannian manifold imitation, we interpret an m-region of the geometric model as a non-isometric embedding of some compact differentiable m-manifold M to $\mathbb{R}^3$. The manifold M can be either closed, i.e. it can be a closed surface or path embedded in $\mathbb{R}^3$, or it can be a manifold-with-boundary, i.e. a surface patch or a block of volume. What is limiting the properties of M in this imitation is that it needs to be embeddable to $\mathbb{R}^3$. That is, its dimension must be less or equal to 3 and for example one cannot imitate non-orientable 3-manifolds within this framework.

Let us denote the map from m-regions to the m-manifold M by $\mu : \mu^{-1}(M) \subset \mathbb{R}^3 \to M$. One can give at least a local coordinate chart (U, x) of M by defining a concrete map

$$\phi_M = x \circ \mu : \quad \mathbb{R}^3 \supset \mu^{-1}(M) \xrightarrow{\mu} M \xrightarrow{x} x(M) \subset \mathbb{R}^m : \tag{5.1}$$

$$\phi_M(\mathbf{p}) = (x^1(\mu(\mathbf{p})), \ldots, x^m(\mu(\mathbf{p}))), \tag{5.2}$$

where $\mathbf{p}$ are points on the preprocessor's coordinate system. The map $\phi_M = x \circ \mu$ is required to be a piecewise diffeomorphism. In the finite element setting, piecewise differentiable maps will suffice as the integration functionals will neglect sets with zero measure.

Now, the m-region in the preprocessor model defines the topology and the discrete representation of the m-manifold M in terms of the finite elements and their connections, but additional charts (U, x) can be defined for the manifold, see Fig. 5.1. The charts (U, x) also provide an alternative representation for the points $p \in M$, i.e. the tuples $\mu(p) = \mathbf{p} \in \mathbb{R}^3$ and $x(p) \in \mathbb{R}^m$ represent the same point $p \in M$. The set $\{(M, \mu^{-1}), (U, x)\}$ is an atlas for the manifold M.

In addition to providing alternative coordinate charts for the m-dimensional regions of the preprocessor model, one can immerse or submerse them in an n-dimensional coordinate chart, which we interpret as the n-manifold N. The dimension n can be larger than 3 and larger or smaller than m. Locally, regions of the preprocessor chart can be given local coordinate charts

$$\phi_N = y \circ \varphi \circ \mu : \quad \mu^{-1}(M) \xrightarrow{\mu} M \xrightarrow{\varphi} N \xrightarrow{y} y(N) : \tag{5.3}$$

$$\phi_N(\mathbf{p}) = (y^1(\varphi(\mu(\mathbf{p}))), \ldots, y^n(\varphi(\mu(\mathbf{p})))) \tag{5.4}$$

The map (y, V) is a chart for the manifold N, and the map φ is now a map between the manifolds M and N, see Fig. 5.1. The map $\phi_N = y \circ \varphi \circ \mu$ is only required to be continuous and piecewise differentiable. That is the minimal requirement for endowing the manifold M with the pullback metric $\varphi^* h$ for producing a Riemannian manifold $(M, \varphi^* h)$ if h is the metric tensor of the Riemannian manifold (N, h).

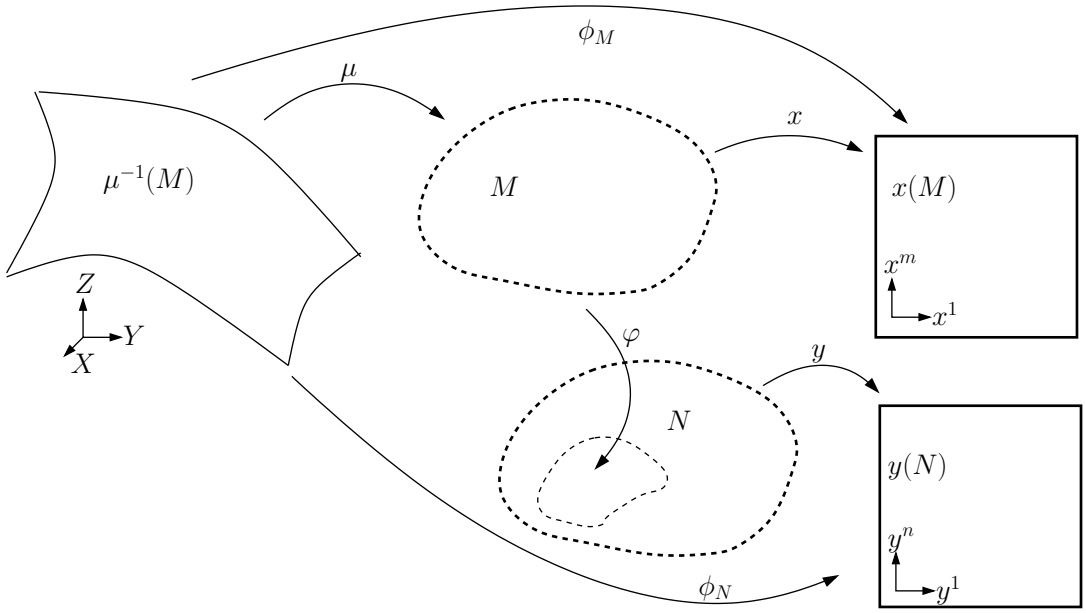


Figure 5.1: Maps from the preprocessor model region to the charts of the manifolds M and N.

5.2.1 Differentials of the charts

Let (U, x) be a chart for the m-manifold M defined via $\phi_M = x \circ \mu : \mu^{-1}(M) \to x(M)$. Then the differential of ϕ_M is an $m \times 3$ matrix $\mathrm{J}_M(\mathbf{p})$

$$\mathrm{J}_M(\mathbf{p}) = \begin{bmatrix} \frac{\partial x^1}{\partial X} & \frac{\partial x^1}{\partial Y} & \frac{\partial x^1}{\partial Z} \\ \vdots & \vdots & \vdots \\ \frac{\partial x^m}{\partial X} & \frac{\partial x^m}{\partial Y} & \frac{\partial x^m}{\partial Z} \end{bmatrix}, \quad \text{where } \mathbf{p} = (X, Y, Z) \in \mu^{-1}(M) \subset \mathbb{R}^3. \tag{5.5}$$

If μ is indeed a piecewise diffeomorphism, the differential $\mathrm{J}_M(\mathbf{p})$ exists almost everywhere in $\mu^{-1}(M)$ and it is surjective with $\operatorname{rank} \mathrm{J}_M(\mathbf{p}) = m$. Further, one of the right inverses of $\mathrm{J}_M(\mathbf{p})$ coincides with the differential of the inverse map $\phi_M^{-1} = \mu^{-1} \circ x^{-1}$ at $\phi_M(\mathbf{p}) = \mathbf{x} \in x(M) \subset \mathbb{R}^m$. We denote the $3 \times m$ -matrix that is the differential of the inverse map by $\mathrm{J}_M(\mathbf{x})^{-1}$, even though it is the actual matrix inverse if and only if $m = 3$. The differentials of the maps ϕ_M and ϕ_M^{-1} enable one to express transformations of real number representations of various tensors between the manifold M chart (U, x) and the preprocessor model coordinate chart μ^{-1}.

Now, $\phi_N = y \circ \varphi \circ \mu : \mu^{-1}(M) \subset \mathbb{R}^3 \to y(N) \subset \mathbb{R}^n$ holds for the continuous and piecewise differentiable map ϕ_N. The differential $\mathrm{J}_N(\mathbf{p}) : \mathbb{R}^3 \to \mathbb{R}^n$ of ϕ_N is an $(n \times 3)$-matrix which is defined similarly as the matrix $\mathrm{J}_M(\mathbf{p})$. Denote the differential of the map $\phi_N \circ \phi_M^{-1}$ by $\mathrm{J}_{NM}(\mathbf{x}) = \mathrm{J}_N(\phi_M^{-1}(\mathbf{x}))J_M^{-1}(\mathbf{x})$. If J_{NM} is surjective, i.e. $\operatorname{rank}(J_{NM}(\mathbf{x})) = n$ holds for almost all $\mathbf{x} \in \chi(M)$, the map φ is a *submersion*. If J_{NM} is injective, i.e. $\operatorname{rank}(J_{NM}(\mathbf{x})) = m$ for each $\mathbf{x} \in \chi(M)$, the map φ is an *immersion* and $\varphi(M)$ is an *immersed submanifold* of N. Further, if $\varphi(M) \subset N$ is *homeomorphic* to M, φ is an *embedding* and $\varphi(M)$ is a *submanifold* of N.

5.2.2 Finite element charts for the manifold

The *Gmsh* preprocessor model consists of m-dimensional regions embedded in $\mathbb{R}^3$, where $0 \leq m \leq 3$. Our goal is to interpret such regions as Riemannian m-manifolds. As the regions are approximated by a mesh, the manifold M is also described by the collection of the m-dimensional finite elements (E^m, ϕ_σ) in the region (see section 2.2.5 on page 37).

To fit this into our framework, we can treat each finite element as a chart (U_σ, u_σ) for for M. The map $\phi_\sigma = (u_\sigma \circ \mu)^{-1} : \mathbb{R}^m \to \mathbb{R}^3$ has the same role as the map $\phi_M^{-1} = (x \circ \mu)^{-1}$ had earlier. Therefore, its differential is $\mathrm{J}_\sigma(\mathbf{u})$. These element-wise charts $(U_\sigma, u_\sigma) : M \to E^m$ can be appended to the atlas of the manifold M, see Fig. 5.2.

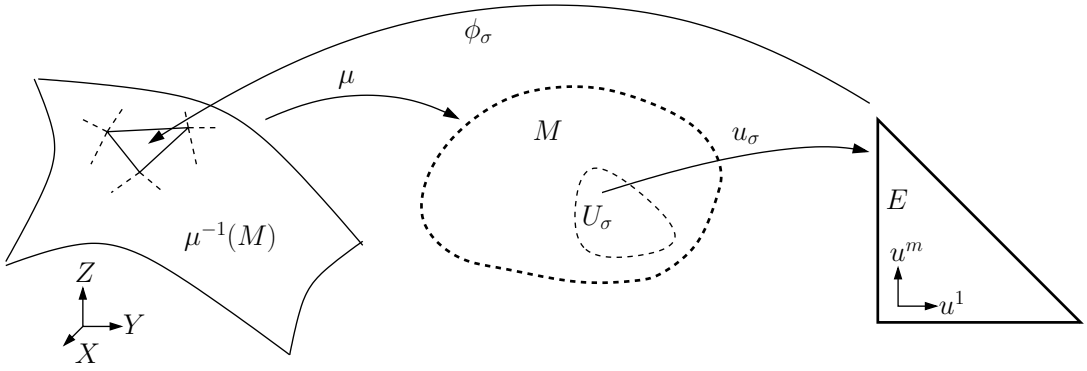


Figure 5.2: Finite elements provide local charts (U_σ, u_σ) for the manifold M.

The differential of the map ϕ_σ^{-1} is given by the pseudoinverse $\mathrm{J}_\sigma^+(\phi_\sigma(\mathbf{u}))$: If $\mathrm{J}_\sigma = \mathrm{U}\Sigma\mathrm{V}^T$ is the singular value decomposition of J_σ, then $\mathrm{J}_\sigma^+ = \mathrm{V}\Sigma^+\mathrm{U}^T$ holds, where the rectangular diagonal matrix Σ^+ is Σ transposed and its non-zero elements are replaced by their reciprocals. The row space of the injective $3 \times m$ matrix $\mathrm{J}_\sigma(\mathbf{u})$ is $\mathbb{R}^m$ and its column space is the m-dimensional tangent space of $\mu^{-1}(M)$ at $\phi_\sigma^{-1}(\mathbf{u})$, which is a subspace of $\mathbb{R}^3$. Then, the equation $\mathrm{J}_\sigma \mathbf{w} = \mathbf{z}$ has a unique solution $\mathbf{w}$ for each tangent vector $\mathbf{z}$ given by $\mathbf{w} = \mathrm{J}_\sigma^+ \mathbf{z}$. For non-tangent vectors $\mathbf{z}$, no solution exists.

In summary, the matrices $\mathrm{J}_\sigma(\mathbf{u})$ and $\mathrm{J}_\sigma^+(\phi_\sigma(\mathbf{u}))$ allows us to represent tensorial objects in terms of the chart u_σ on each submanifold M_σ of the m-manifold M. The advantage of such representation is that when $m < 3$ holds, the coefficient vector of a tensor has less components than in the embedding chart μ^{-1} representation.

5.2.3 Chains

In the finite element setting, k-chain can be considered as an integer combinations of k-dimensional finite elements $\sigma^k = (E^k, \phi_\sigma)$. Therefore, the differentials J_σ of the maps $\phi_\sigma : E^k \to \mathbb{R}^3$ can be used to construct the pullback of a differential k-form to a k-chain, needed in the integration, see Fig. 5.3.

5.2.4 Implementation details

The object-oriented `C++` implementation is based on the *Gmsh* API and its mesh data structure. The domain is divided into m-regions called "physical groups" in *Gmsh* jargon,

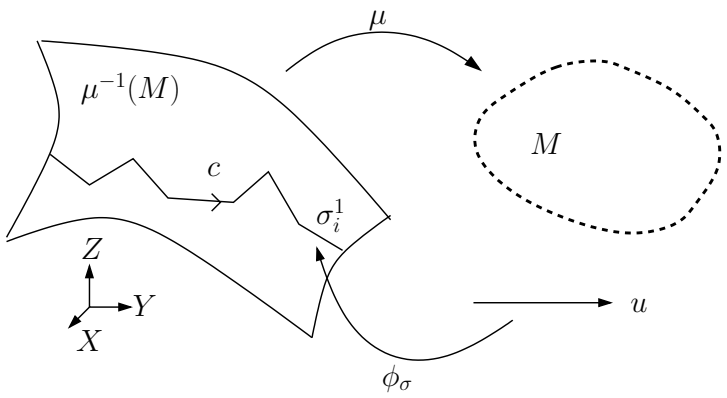


Figure 5.3: The pullback maps $(\mu \circ \phi_\sigma)^* : \Omega^1(M) \to \Omega^1(\mathbb{R})$ can be used pullback a differential 1-form ω on M to the Euclidean reference cell E^1 of a finite element $\sigma = (E^1, \phi_\sigma)$ that belongs to a 1-chain c. Such construction can be used to compute the integral $\int_c \omega = \sum_i c^i \int_{\sigma_i^1} \omega$.

covered with m-dimensional mesh elements. From these, the user can instantiate an object of the Riemannian manifold class that interprets the region as an m-manifold (M, g).

Points

The points p of the m-manifold M are imitated as pairs $(\sigma^m, \mathbf{u})$ of a finite element $\sigma = (E^m, \phi_\sigma)$ and a point $\mathbf{u} \in E^m \subset \mathbb{R}^m$. The reason for this is that we want to separate overlapping points on the boundaries of the mesh elements in order to be able to have discontinuities of fields across element interfaces. The model coordinates of a point $(\sigma^m, \mathbf{u})$ are $\mathbf{p} = \phi_\sigma(\mathbf{u}) \in \mathbb{R}^3$.

The points $(\sigma_1^m, \mathbf{u}_1)$ and $(\sigma_2^m, \mathbf{u}_2)$ are considered as different points even though $\mathbf{p}_1 = \phi_{\sigma_1}(\mathbf{u}_1) = \phi_{\sigma_2}(\mathbf{u}_2) = \mathbf{p}_2$ would hold. That is, we define the equivalence relation between the points $p_1 = (\sigma_1^m, \mathbf{u}_1)$ and $p_2 = (\sigma_2^q, \mathbf{u}_2)$ to be

$$p_1 \sim p_2 \iff \sigma_1^m = \sigma_2^q \text{ and } \mathbf{u}_1 = \mathbf{u}_2. \tag{5.6}$$

We have that $(p_1 \sim p_2) \implies (\mathbf{p}_1 = \mathbf{p}_2)$ but the reverse implication does not hold, as overlapping points may be based on different finite elements. Later, we refer to $\mathbf{p} = \mu^{-1}(p) \in \mathbb{R}^3$ as the **model coordinates** of a point p, and to $\mathbf{x} = x(p) \in \mathbb{R}^m$ as the **manifold coordinates** of the point on an m-manifold M with a chart (U, x).

Manifold and charts

The instantiated Riemannian manifold object (M, g) is aware of by which finite elements it is converted and by default, it has a partial atlas $(U_\sigma, u_\sigma)\}$ of charts described in the section 5.2.2 on page 106. However, the user can define an another atlas $\{(M, x)\}$ for the manifold M by providing a piecewise diffeomorphism $\phi_M = x \circ \mu : \mathbb{R}^3 \supset \mu^{-1}(M) \to x(M)$ and the differentials J_M and J_M^{-1}. The map ϕ_M is implemented as an abstract class, whose derived classes are eligible to be used as charts for the manifold M. One can also construct the maps ϕ_M from previous computational results.

The class representing ϕ_M provides methods to obtain representation $\mathbf{x}$ of a point $p \in M$ on a chart of M and the representation $\mathbf{p}$ in the model coordinates, as well as the differentials $\mathrm{J}_M(\mathbf{p})$ and $\mathrm{J}_M^{-1}(\mathbf{x})$ at a given point p. From the differentials, one can construct pullbacks and pushforwards for the various tensors on M, as will be described in a later section.

The model coordinates of a point $p = (\sigma^m, \mathbf{u}) \in M$ are $\mu^{-1}(p) = \mathbf{p} = \phi_\sigma(\mathbf{u})$ and the manifold coordinates are $x(p) = \mathbf{x} = \phi_M(\mathbf{p}) = \phi_M(\phi_\sigma(\mathbf{u}))$. Given a point $\mathbf{p} \in \mathbb{R}^3$ in the model coordinates, it can be turned into point $p \in M$ using a two step algorithm: First, find all finite elements near that point in $\mathbb{R}^3$ using an octree search [46]. Second, if a found finite element σ^m belongs to M, we have $p = (\sigma^m, \phi_\sigma^{-1}(\mathbf{p}))$. The result is not unique since there can be other finite elements near $\mathbf{p}$ that belong to M.

Maps between manifolds

Let Q be a q-manifold based on an q-region of a preprocessor model, and let $\mu : \mathbb{R}^3 \to M$ and $\kappa : \mathbb{R}^3 \to Q$ be the corresponding maps from the model coordinates to the manifolds. If their preprocessor representations overlap, i.e. if $\mu^{-1}(M) \cap \kappa^{-1}(Q) \neq \emptyset$ holds, we can construct a map $\kappa \circ \mu^{-1} : M \to Q$, its differential and possibly its inverse map and the differential of its inverse. Such map is needed for example if one wanted to compute a trace of a differential form on M to a submanifold $Q \subset \partial M$. We have the following two possibilities.

1. The mesh element σ^m of $p = (\sigma^m, \mathbf{u}) \in M$ also belongs to the manifold Q. Then simply $(\kappa \circ \mu^{-1})(p) = p = (\sigma^m, \mathbf{u})$. In this case, $m = q$ must hold. The differential is $\mathrm{J}_Q(\kappa^{-1}(p))\mathrm{J}_M^{-1}(x(p))$.

2. The mesh element σ^m of $p = (\sigma^m, \mathbf{u}) \in M$ does not belong to Q. Then, we map $\mathbf{p} = \phi_\sigma(\mathbf{u}) \in \mathbb{R}^3$ and find all mesh elements near[1] that point in $\mathbb{R}^3$ using an octree search [46]. If a found mesh element σ^q belongs to Q, we map $p' = (\kappa \circ \mu^{-1})(p) = (\sigma^q, \phi_{\sigma^q}^{-1}(\mathbf{p}))$. The differential is $\mathrm{J}_Q(\kappa^{-1}(p'))\mathrm{J}_M^{-1}(x(p))$.

In the latter case, the two manifolds M and Q may be based on entirely different preprocessor models, i.e. they may have different, but overlapping finite element meshes in $\mathbb{R}^3$. The relation between their points is established via the equivalence of 3-tuples $\mathbf{p}$ of $\mathbb{R}^3$.

5.3 Imitation of tensorial objects

Tensorial objects are coordinate-free in a sense that although their representations depend on the used coordinate system, any coordinate system can be used to perform the computations. Since tensors are linear maps, their coordinate transformations must be linear maps as well to conform to the above property. Furthermore, tensors are either *covariant*, *contravariant*, or mixture of both properties depending whether their arguments are k-covectors or k-vectors or both.

In this section, we present how the most usual contravariant and covariant tensors are represented and implemented in the Riemannian manifold interface. Namely, k-covectors, the metric tensor, and k-vectors.

[1] To a tolerance that stems from the floating-point number inaccuracy.

5.3.1 Representation and transformations of tensors on the manifold

On a local coordinate chart (U, x) of the m-manifold M, tangent vectors $v_p \in T_pM$ and covectors $\omega_p \in T_p^*M$ have coefficient representations $\mathbf{v} = [\mathbf{v}^1, \ldots, \mathbf{v}^m]^T$ and $\boldsymbol{\omega} = [\boldsymbol{\omega}_1, \ldots, \boldsymbol{\omega}_m]^T$ in $\mathbb{R}^m$ with respect to the coordinate basis $\{\partial/\partial x^1, \ldots, \partial/\partial x^m\}$ and the coordinate cobasis $\{\mathrm{d}x^1, \ldots, \mathrm{d}x^m\}$ of (U, x). That is, with the representation $\boldsymbol{\omega}$, we have $\omega_p = \sum_{i=1}^m \omega_i \mathrm{d}x^i$.

While the representation $\boldsymbol{\omega}$ depends on the coordinate patch, the linear combinations $\omega_p = \sum_{i=1}^m \omega_i \mathrm{d}x^i$ is a coordinate-free object, since the "differentials" $\mathrm{d}x^i$ can be naturally expanded by the *chain rule* of elementary calculus to conform with a different coordinate chart.

Recall from section 2.2.3 on page 31 that if $\mathbf{v}'$ and $\mathbf{v}$ are the representations of a tangent vector v_p on overlapping charts (U', x') and (U, x), they are related by $\mathbf{v}' = \mathrm{J}\mathbf{v}$, where J is the differential of the map $x' \circ x^{-1}$. Let $\boldsymbol{\omega}'$ and $\boldsymbol{\omega}$ similarly represent a tangent covector ω_p. It was required that $\boldsymbol{\omega}'^T\mathbf{v}' = \boldsymbol{\omega}'^T\mathrm{J}\mathbf{v} = (\mathrm{J}^T\boldsymbol{\omega}')^T\mathbf{v} = \boldsymbol{\omega}^T\mathbf{v}$ hold. Therefore, the matrix J^T performs the transformation for the tangent covectors. Similarly, if φ is a piecewise differentiable map $\varphi : M \to N$, one can pushforward a tangent vector v_p to the manifold N, denoted $\varphi_* v_p$, and pullback a tangent covector ω_p to the manifold M, denoted $\varphi^* \omega_p$.

The exterior and tensor algebras of the tangent vectors and the covectors induce the vector spaces of k-(co)vectors and (k, l)-tensors via the exterior product $\wedge$ and the tensor product $\otimes$, respectively, as discussed in the section 2.1.4 on page 27. We denote the spaces of (k, l)-tensors by $\bigotimes_k^l(T_pM)$ and of k-covectors by $\Lambda^k(T_p^*M) \subset \bigotimes_0^k(T_pM)$ and k-vectors $\Lambda^k(T_pM) \subset \bigotimes_k^0(T_pM)$, respectively. The dimension of the spaces on an m-manifold are $r = \binom{m}{k}$ for k-(co)vectors and $r = m^k m^l$ for (k, l)-tensors.

The representations of k-(co)vectors can be given in the multi-index notation. For example, the representation of k-covector is $\omega_p = \sum_{\bar{I}} \boldsymbol{\omega}_I \mathrm{d}x^I$, where $\mathrm{d}x^I = \mathrm{d}x^{i_1} \wedge \ldots \wedge \mathrm{d}x^{i_k}$. The pushforward and the pullback together with the exterior product and the tensor product also induce the transformation rules for k-(co)vectors and (k, l)-tensors, see section 2.1.4.

A (k, l)-tensor field on the manifold M is a map $M \to \bigotimes_k^l(TM)$, where $\bigotimes_k^l(TM)$ denotes the (k, l)-tensor bundle of the manifold M. That is, a tensor field is required to associate to each point $p \in M$ a representation of a point-wise tensor.

5.3.2 Metric tensor on the manifold

The metric tensor g is symmetric $(0, 2)$ tensor that has a special meaning on the Riemannian manifold (M, g). With the metric tensor one can compute inner products of other tensors and transform (k, l) tensors to $(k - 1, l + 1)$ and $(k + 1, l - 1)$ tensors. Especially, it can be used to to compute the Hodge operator and flat and sharp operators of k-(co)vectors.

The metric tensor $g : M \to \bigotimes_0^2(T_pM)$ of the Riemannian m-manifold (M, g) is point-wise represented by a symmetric $m \times m$-matrix G. The metric tensor at $p \in M$ is a map $g_p : T_pM \times T_pM \to \mathbb{R}$. Thus, the coefficients with respect to the coordinate basis are $\mathrm{G}_{ij} = g_p(\partial/\partial x^i, \partial/\partial x^j)$. The representations G and G′ of g on charts (U, x) and (U', x')

are related by $\mathrm{G}' = \mathrm{J}^T \mathrm{G} \mathrm{J}$.

There's a variety of methods one can define the metric tensor on M. Most commonly, one can pullback the Euclidean metric from the preprocessor coordinates to M using the differential of the map μ. One can also provide its coefficients as an analytical expressions on the model coordinates or on the chart coordinates of M, or define a result of a previous numerical computation to provide the coefficients. Lastly, one can define an another, high dimensional auxiliary Riemannian manifold (N, h) and use the pullback metric tensor $g = \varphi^* h$ on M. Again, h can be either the Euclidean metric tensor or its coefficients can be defined via analytical expressions or by previous numerical computations.

5.3.3 Implementation details

The k-(co)vectors, metric tensors and general (k, l)-tensors and tensor fields are handled as coordinate-independent objects that live on the manifold M. That is, the user manipulates the coordinate-free mathematical objects, rather than their real number representations. Under the hood, the real number representations of the objects are always accompanied with the information about the manifold and chart on which they apply. The user needs to pay attention to the representations only when inputting or outputting data. Currently, the real number representations are with respect to the coordinate basis of the chart.

Tensors and tensor fields

The implementation of the tensor class in the interface depends heavily on `C++` templates. `C++` programming language supports so-called *non-type template arguments*. For example, a template argument can be a specific integer or a member in an enumeration, rather than an integer or *enumeration* type. In the implementation, the tensor class header reads:

```
template <int n, int k, int l, TensorSymmetry s> class Tensor
```

Here `TensorSymmetry` is an enumeration type for various symmetries a tensor might posses, for example skewsymmetry. Integers `n`, `k`, and `l` are the dimension of the manifold, and contravariant and covariant indices of the tensor, respectively. With this implementation, the specific type of the tensor is known at the *compile time*. An alternative implementation would be to store the type information as member variables of the class. Then, the type of the tensor would be only known at *run time*. The static (compile time) type information has several advantages:

1. The number of coefficients that represent the tensor on a chart of a manifold is known at compile time. Therefore, static data structures can be used to store the coefficients. This increases the performance in computations.

2. The specific algorithm to perform some tensor operation is chosen at compile time by the compiler. This removes a performance overhead when the algorithm is chosen at run time. For example, the algorithm to perform the inner product of k-covectors depends on the manifold dimension and the covariant index k.

3. When performing operations on tensors, in addition to computing the coefficients, the type of the resulting tensor also needs to be computed. Our template implementation performs the type computation at compile time. This is again a performance gain.

4. The `C++` compiler static type checking guarantees that the performed operations of tensors are legitimate. That is, some programming errors are caught at compile time rather than at run time.

5. Compiler optimizations are more efficient with static type information.

However, the static type information has one considerable disadvantage. If the program to be written is such that the types of the tensors are only known at run time, extensive *conditional constructs* must be written in some places of the program to select the appropriate tensor type.

The field class in this interface is templated by the tensor type, for example k-(co)vectors or symmetric $(0,2)$-tensors. It is an abstract class whose derived classes are either discrete or analytic. A discrete field stores the representations of tensors at each vertex of each mesh element σ in the mesh, and interpolates the element at other points using Whitney 0-forms. Discrete fields can be constructed during computations. An analytic field evaluates a user-defined expression at a given point. In addition, the user can derive a class whose behaviors he has a complete command and responsibility. The abstract base class provides the interface to use it within the Riemannian manifold library.

Operations on tensors and tensor fields

The interface hides the coefficient vector computations of the operation on tensors by providing functions whose inputs and outputs are objects rather than real number representations. Point-wise, i.e. algebraic operations on tensor fields can be performed using the representations of various tensor at the given points.

In contrast, the analytic operations on a tensor field need some information around its behavior in a neighborhood, and therefore cannot be performed for tensor fields on the manifold N, since the map $\varphi : M \to N$ is only required to be piecewise differentiable. However, one can pullback covariant tensor fields on the manifold N to the manifold M and perform analysis for the pullback fields.

Gmsh API provides the finite element σ shape functions λ_i of any order and their partial derivatives $\frac{\partial \lambda_i}{\partial u^j}$ on the reference m-cell E^m, the maps $\phi_\sigma : E^m \to \phi_\sigma(E) \subset \mathbb{R}^3$ and their differentials J_σ, and Gauss integration data. Therefore, the exterior derivative, integration, and trace of differential k-forms ω on the manifold M are implemented via the pullback $(\phi_M \circ \phi_\sigma)^* \omega$ on the reference cell E^m.

The exterior derivative $\mathrm{d}\omega$ [24] of a differential k-form can be computed element-wise using the partial derivatives of the shape functions λ_i, once given the k-covectors $\omega_i^\sigma = \sum_J \boldsymbol{\omega}^i{}_J \mathrm{d}u^J$ on each vertex i of an element σ on the element-wise chart (U_σ, u_σ). Let $\mathrm{d}\lambda_i = \sum_{j=1}^m \frac{\partial \lambda_i}{\partial u^j} \mathrm{d}u^j$ hold, then $\mathrm{d}\omega = \sum_i (\mathrm{d}\lambda_i \wedge \omega_i^\sigma)$ holds.

A differential k-form ω on the m-manifold M can be integrated over k-dimensional finite elements $\sigma^k = (E^k, \phi_\sigma)$ in a finite element environment. The integral can be

evaluated by the change of variables formula to be

$$\int_{\sigma^k} \omega := \int_{u_\sigma^{-1}(E^k)} \omega = \int_{E^k} u_\sigma^{-*}\omega = \int_{E^k} \mathrm{t}\omega. \tag{5.7}$$

The integration over each element is carried out on the pullback with the Gauss quadrature rules on the Euclidean reference k-cell E^k. Then, the pairing $((\phi_M \circ \phi_\sigma)^*\omega_p)(w_p) \in \mathbb{R}$ is evaluated at Gauss integration points, where the k-vector w_p is the Gauss integration weight at the point p. The procedure is also extended to the integration of k-forms over *k-chains*, formal sums of k-dimensional finite elements as discussed earlier.

Other differential operators on tensor fields require an additional structure called *Connection* on the manifold [24]. That is, mere coordinate derivatives of their component functions are not adequate. Without Connection, a result of a tensor differentiation would not be, in general, a tensor.

Such differential operators are needed extensively in many fields of physics, such as elasticity and fluid dynamics, but not in traditionally in electromagnetics. Since the scope of this thesis is mainly in electromagnetics, the Connection is not currently implemented in the interface. However, this could be done in a similar fashion how the metric tensor is defined by providing its component functions on some chart of the manifold. For a Connection, one could provide its *connection coefficients* [24] on some chart of the manifold.

5.4 Application examples

5.4.1 Example program: Poisson equation and harmonic field solver on Riemannian 3-, and 2-manifolds

This program solves two types of boundary value problems for a scalar field on an orientable Riemannian 3-, or 2-manifold (M, g) that is represented as a meshed region in the *Gmsh* preprocessor. This program is utilized in the examples 5.4.2, 5.4.3, 5.4.4, and 5.4.5.

The first problem is the Poisson boundary value problem with Dirichlet and Neumann boundary conditions. That is

$$\mathrm{d}\star\kappa\mathrm{d}\alpha = \eta \tag{5.8}$$
$$\mathrm{t}_S\alpha = \beta \tag{5.9}$$
$$\mathrm{t}_{S^*}\star\mathrm{d}\alpha = \gamma \tag{5.10}$$

The second problem is to solve for a harmonic 1-form ω by using a local scalar potential α and cohomology basis functions:

$$\mathrm{d}\omega = 0 \tag{5.11}$$
$$\mathrm{d}\star\kappa\omega = 0 \tag{5.12}$$
$$\mathrm{t}_S\omega = 0 \tag{5.13}$$
$$\mathrm{t}_{S^*}\star\omega = 0 \tag{5.14}$$

With cohomology conditions

$$\int_{z_i^1} \omega = A_i, \text{ where } [z_i^1] \in H_1(M, S), \tag{5.15}$$

$$\int_{z_i^{n-1}} \star\kappa\omega = B_i, \text{ where } [z_i^{n-1}] \in H_{n-1}(M, S^*) \tag{5.16}$$

of which only $\beta_1(M, S) = \beta_2(M, S^*)$ are independent as discussed in the previous chapter.

The input for the program is a Riemannian n-manifold (M, g) object, $n \leq 3$, together with a tensor field $\kappa : M \to \bigotimes_1^1(TM)$. For the first problem, the input also includes differential forms $\eta : M \to \Lambda^n(T^*M)$, $\beta : S \to \Lambda^0(T^*S)$, and $\gamma : S^* \to \Lambda^{n-1}(T^*S^*)$. For the second problem, the input includes cohomology basis functions $\zeta^i : M \to \Lambda^1(T^*M)$ such that $[\zeta^i] \in H^1_{\mathrm{dR}}(M, S)$ together with coefficients A_i or B_i.

The solver is implemented as a `C++` function that assembles the linear system of the boundary value problem. The function has a template argument that is an integer equal to the dimension n of the manifold M. The function performs a typical finite element assembly procedure, however with a twist that it uses the Riemannian manifold interface objects to compute the integrands.

The resulting assembler code is independent from the dimension n, the metric tensor g, and the coordinate chart (M, x) of the Riemannian n-manifold (M, g) on which the boundary value problem was set up. To demonstrate this, the code that computes the expressions $\star\kappa \mathsf{d}\mathsf{n}^i \wedge \mathsf{d}\mathsf{n}^j$ at an integration point of a mesh element is the following:

```
// Loop over the mesh element vertices
for(int i = 0; i < me->getNumVertices(); i++) {

  // Apply kappa and Hodge operator to the exterior derivative of a
  // of a nodal shape function at node i evaluated at an integration point
  Covector<n,n-1> ai = h(kappa*dn.at(i));

  // Loop over the mesh element vertices
  for(int j = 0; j <= i; j++) {

    // Compute the wedge product with the exterior derivative of a
    // of a nodal shape function at node j evaluated at an integration point
    Covector<n,n> aij = ai % dn.at(j);

    // Add the result to the element-wise matrix
    // multiplied by the integration weight
    localMatrix(i, j) += aij(0)*weight;
  }
}
```

where the function `h` computes the Hodge operator and `%` performs the wedge product. The Riemannian manifold library translates the above lines to the actual numerical computations which depend on the dimension n, on the metric tensor g and on the chart (M, x).

The result for both problem types is the 0-form $\alpha = \sum_{\sigma_i^1 \in M \setminus S} a_i \mathsf{n}^i$. Then in the second problem type,

$$\omega = \mathrm{d}\alpha + \sum_{i=1}^{\beta_1(M,S)} A_i \mathsf{E}^i \tag{5.17}$$

holds, where the cohomology basis functions E are constructed from the basis elements of $H^1(M, S)$.

5.4.2 Parametrization for a surface patch

In this example, a surface patch in the preprocessor model interpreted as a Riemannian 2-manifold (M, g). It is endowed with the element-wise chart (M, x), $x = \sum_\sigma u_\sigma$. The metric tensor $g = \mu^{-*}\delta$, the pullback of Euclidean metric from the preprocessor coordinates to the surface. Then, a surface parametrization is obtained by solving two Laplace problems on the manifold M for two coordinate functions $\xi_1, \xi_2 : M \to \mathbb{R}$. Those result a function

$$\phi_M = (\xi_1 \circ \mu, \xi_2 \circ \mu) : \mathbb{R}^3 \to \mathbb{R}^2 \quad \text{so that} \tag{5.18}$$

$$\phi_M(x, y, z) = (\xi_1(\mu(x, y, z)), \xi_2(\mu(x, y, z))) \in [0, 1] \times [0, 1] \subset \mathbb{R}^2 \tag{5.19}$$

for all (x, y, z) that belong to the surface patch in the preprocessor model. The map ϕ_M is then used to obtain an another chart ξ for the Riemannian 2-manifold (M, g). Fig. 5.4 visualizes the surface, the chart ξ, and the metric tensor g.

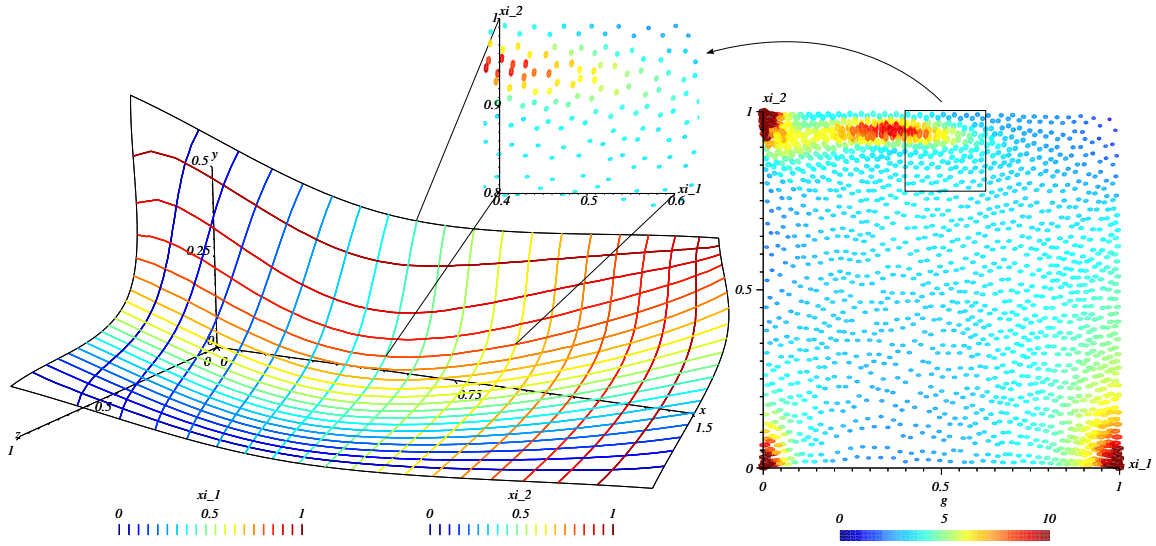
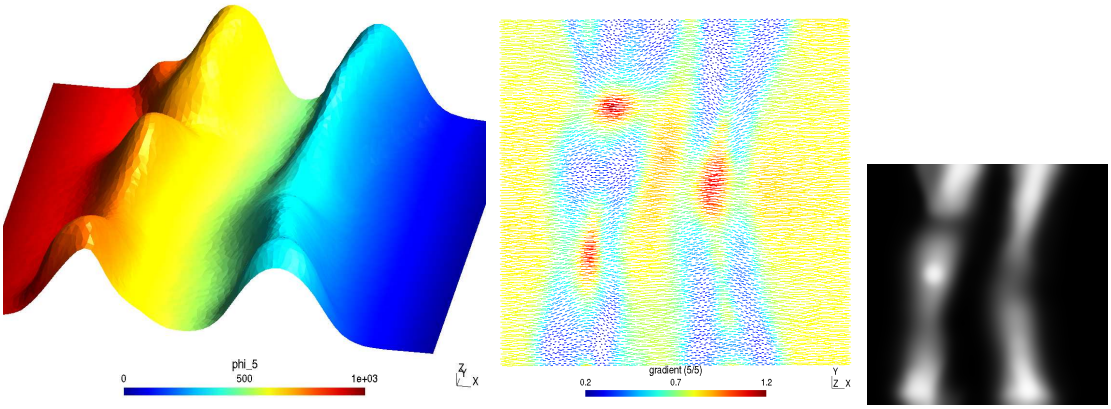


Figure 5.4: A chart ξ for a surface patch M and the metric tensor g visualized as metric unit balls on the chart, i.e. how the parametrization "stretches" the surface.

5.4.3 Laplace problem with a pullback metric

In this example, a preprocessor coordinate model region $[0, 1] \times [0, 1] \subset \mathbb{R}^2$ is interpreted as a Riemannian 2-manifold (M, g). The metric tensor g is obtained by pulling back the

Euclidean metric from a 3-manifold N, which is related to the manifold M by the map $\varphi = \phi_N \circ \mu^{-1}$, where $\phi_N(x, y, z) = (x, y, h(x, y))$ and h is a height map given by a bitmap in Fig. 5.5 (c)



(a) The solution field f and the image of M under the map $\phi_N \circ \mu^{-1}$.

(b) The vector field ∇f on a chart (M, x) of the manifold M.

(c) The height map h used to define the map ϕ_N.

Figure 5.5: Laplace problem for f solved on a Riemannian manifold (M, g) using the pullback metric $g = (\phi_N \circ \mu^{-1})^* \delta$.

The manifold M is given a chart (M, x) defined by $x = \phi_M \circ \mu^{-1}$, where $\phi_M(x, y, z) = (x, y)$. The matrix representation of the pullback metric $g = (\phi_N \circ \mu^{-1})^* \delta$ on the chart (M, x) is $J_N^T J_N$. Fig. 5.5 (a) and (b) show the solution for the Laplace problem $\mathrm{d}\star\mathrm{d}f = 0$ of the scalar field f and its gradient $\nabla f = \mathrm{d}f^\sharp$ on the chart (M, x). The boundary conditions are $\mathrm{t}f = 0$ and $\mathrm{t}f = 1000$ on two sides and $\mathrm{t}\star\mathrm{d}f = 0$ on the others.

5.4.4 Invisibility cloaking in electrostatics

In this example we consider a setting where one has placed blob of dielectric material between capacitor plates, see Fig. 5.6. One is interested to find such anisotropic permittivity for the rest of the material between the plates that "hides" the dielectric blob from the electric field. That is, to make it seem that the there's nothing but air between the plates when making capacitance measurements for the capacitor. That is, the *Dirichlet-to-Neumann map* [8] is the same if there was no dielectric blob.

Let 3-manifold M denote the volume between the capacitor plates. The manifold M is endowed with a global coordinate chart (M, x), so that $x(\mu(\mathbf{p})) = \mathbf{p}$, i.e. an identity map from the preprocessor model coordinates.

We solve the problem by finding a new coordinate chart (M, y) for the Riemannian manifold (M, g) which diminishes the dielectric blob. In practice, the blob is mapped to a sphere of small radius. Then, we ***set*** the metric tensor g of M to be such that its matrix representation $\mathrm{G}_y = \mathrm{I}$ is the identity matrix on the chart (M, y). If J_{xy} is the differential of the map $y \circ x^{-1}$, then on the chart (M, x) the metric tensor has a representation $\mathrm{G}_x = \mathrm{J}_{xy}^T \mathrm{J}_{xy}$.

The required anisotropic permittivity $\underline{\epsilon}$ can be obtained from the matrix G_x. Let $e = e_1 \mathrm{d}x^1 + e_2 \mathrm{d}x^2 + e_3 \mathrm{d}x^3$ denote the electric field on M, where $\mathbf{e}_x = \begin{bmatrix} e_1 & e_2 & e_3 \end{bmatrix}^T$ is its

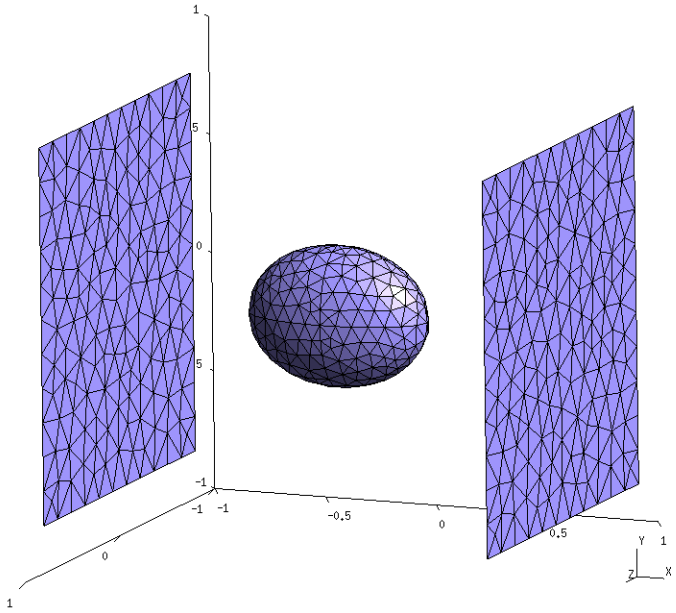


Figure 5.6: Dielectric blob between two capacitor plates.

representation on chart (M, x). Then, the electric flux density $d = d_1 \mathrm{d}x^2 \wedge \mathrm{d}x^3 + d_2 \mathrm{d}x^3 \wedge \mathrm{d}x^1 + d_3 \mathrm{d}x^1 \wedge \mathrm{d}x^2 = \epsilon_0 \star e$ has the representation $\mathbf{d}_x = \begin{bmatrix} d_1 & d_2 & d_3 \end{bmatrix}^T = \epsilon_0 \mathrm{G}_x^{-1} \sqrt{\det \mathrm{G}_x} \mathbf{e}_x$ on the chart (M, x), where ϵ_0 is the vacuum permittivity. Therefore, a material that has anisotropic permittivity $\underline{\epsilon} = \epsilon_0 \mathrm{G}_x^{-1} \sqrt{\det \mathrm{G}_x}$ cloaks the dielectric blob. In other words, $\underline{\epsilon}$ realizes such Riemannian manifold (M, g) where there seems to be nothing but air between the capacitor plates.

The solution for G_x, hence for $\underline{\epsilon}$, that cloaks the dielectric blob is not unique. Any diffeomorphic transition function $y \circ x^{-1}$ that maps the dielectric blob to a sphere of small radius can be used to induce the pullback metric matrix G_x. One such $y \circ x^{-1}$ can be obtained by solving the following three Laplace boundary value problems.

Denote by $M_b \subset M$ the dielectric blob. Let $x(M) = [-1, 1] \times [-1, 1] \times [-1, 1] \subset \mathbb{R}^3$ hold. We solve the following three problems for y^i, $i = 1, 2, 3$ on (M, g):

$$\mathrm{d} \star \mathrm{d} y^i = 0, \tag{5.20}$$

$$\mathrm{t} y^i = x^i \text{ when } x^i = -1 \text{ or } x^i = 1, \tag{5.21}$$

$$\mathrm{t} \star \mathrm{d} y^i = 0 \text{ elsewhere on } \partial M, \text{ and} \tag{5.22}$$

$$\mathrm{t}_{\partial M_b} y^i = \begin{cases} r \sin\theta \cos\phi & \text{when } i = 1 \\ r \sin\theta \sin\phi & \text{when } i = 2 \\ r \cos\theta & \text{when } i = 3 \end{cases}, \tag{5.23}$$

where r is the radius of the small sphere and θ and ϕ are the polar and the azimuthal angles from the origin to a point (x^1, x^2, x^3) on the surface ∂M_b of the dielectric blob. Any metric tensor g can be used to solve for the coordinate functions $y^i \circ x^{-1}$. A different choice produces a different diffeomorphism $y \circ x^{-1}$.

In Fig. 5.7 is the solution of the Laplace problem $\mathrm{d} \epsilon \star \mathrm{d} \varphi = 0$ the Riemannian manifold (M, g), represented on two charts, (M, x) and (M, y). On the chart (M, y), the represen-

tation G_y of the used metric tensor is the identity matrix. On the chart (M, x) where the computation takes place, the metric tensor has the representation $\mathrm{G}_x = \mathrm{J}_{xy}^T \mathrm{J}_{xy}$.

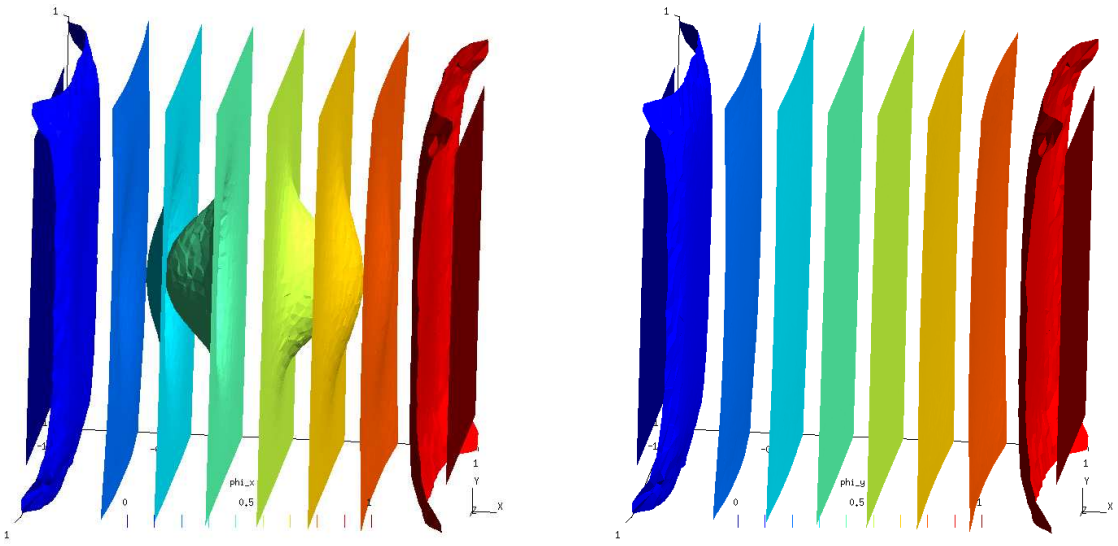


Figure 5.7: Solution φ for the Laplace problem between the capacitor plates on the chart (M, x), left, and on the chart (M, y), right. As the dielectric blob is indistinguishable for the outside observers, the potential φ is a constant inside the blob.

The actual scientific and engineering problem in such cloaking arrangement is to manufacture a material that has the desired location dependent anisotropic permittivity $\underline{\epsilon}$. Therefore in practice, one might want to solve for such $y \circ x^{-1}$ that takes the manufacturability of the material into account. One might for example solve for such $y \circ x^{-1}$ that minimizes the off-diagonal terms of $\underline{\epsilon}$, instead of the one produced by the above Laplace problems.

5.4.5 Finding an atlas of charts for a surface

This example combines cohomology computations to the Riemannian manifold library. We consider an ***orientable*** surface embedded in $\mathbb{R}^3$ as a compact 2-manifold M. The goal is to find an atlas $\mathcal{D} = \{(U_i, x_i)\}$, where $x_i \circ \mu : \mu^{-1}(M) \to x_i(M)$, of coordinate charts for M whose domains U_i cover M, i.e. $\bigcup_i U_i = M$.

The chart images $x_i(M)$ are wanted to be Cartesian products of real number intervals, i.e. $x_i(M) = [a, b] \times [c, d] \subset \mathbb{R}^2$ holds. Such rectangular charts are convenient to work with in many applications. In the general case, a single such 2-dimensional coordinate chart cannot describe the whole surface, since the surface may be closed and/or have handles or holes.

The representatives of the basis elements of the cohomology spaces $H^1(M)$ and $H^1(M, \partial M)$ can be used as "zippers" to divide the triangulated surface into patches which are then provided with 2-dimensional coordinate charts. The coordinate functions will be the associated to the harmonic representatives of the basis elements of the corresponding de Rham cohomology spaces.

As an example, consider the surface depicted e.g. in Fig. 5.8 which constitutes the 2-manifold M. Since the surface has a "branch", it cannot be parametrized with just

two rectangular charts as one could parametrize a surface of a cylinder. To find an atlas for this example 2-manifold, we concentrate on finding $2 = \beta_1(M) = \beta_1(M, \partial M)$ charts for it, while the still missing charts can be constructed from those. The four basis elements of $H^1(M)$ and $H^1(M, \partial M)$ can be used to construct four coordinate maps $x^i, y^i : \mathbb{R}^3 \to \mathbb{R}$ and pairs of them will constitute coordinate charts $\phi_1, \phi_2 : \mathbb{R}^3 \to \mathbb{R}^2$, where $\phi_i = (x_i \circ \mu, y_i \circ \mu)$ holds.

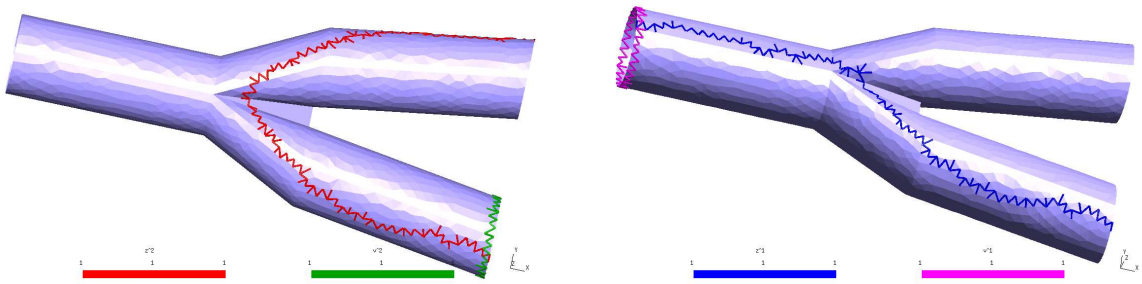


Figure 5.8: Representatives of the cohomology basis elements. The cochains on the left are used to construct the chart (U, x) and the cochains on the right are used to construct the chart (V, y).

In Fig. 5.8 we depict the basis elements of $H^1(M)$ and $H^1(M, \partial M)$, and in Fig. 5.9 we depict two coordinate charts solved using the example program 5.4.1.

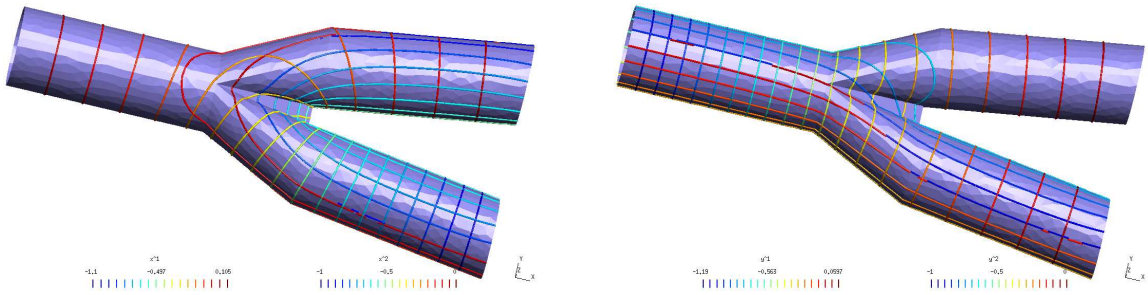


Figure 5.9: Equi-coordinate lines of the coordinate charts x and y for the 2-manifold M. The chart (U, x) covers the two branches on the right, while the chart (V, y) cover the lower branches. The charts have a discontinuity along the basis element representatives of $H^1(M)$. Therefore, two additional charts would be needed to constitute a complete atlas for the manifold M.

5.4.6 Eddy currents on a conductor surface

In eddy current problems, the skin effect may restrict the eddy currents to a very thin layer on the conductor surface. Then, one can approximate that the eddy currents are tangential to the surface and constant with respect to the distance from the surface inside the skin depth region.

In this example, we compute eddy currents on a conductor surface embedded in $\mathbb{R}^3$ as a problem on Riemannian 2-manifold (S, g_s). The benefit of this approach is that the

conducting region is treated as purely two dimensional. Therefore, a mere scalar-valued potential is sufficient to describe the surface current density.

In order to give the emphasis to the Riemannian manifold implementation rather than to the cohomology, we assume that the surface S is closed and that the surface has no handles. That is, $\partial S = \emptyset$ and $\beta_1(S) = 0$ hold.

The non-conducting regions outside and inside the conducting surface are denoted M_1 and M_2, respectively, and they are interpreted as 3-manifolds with the Cartesian metric. The metric tensor on S is the pullback of the Cartesian metric, induced by the embedding map: $\varsigma : \mathbb{R}^3 \supset \varsigma^{-1}(S) \to S$.

On the manifolds M_i we look for a pair of harmonic fields $(b_i, h_i) \in \mathcal{H}^2(M, S_b) \times \mathcal{H}^1(M, S_h)$ which satisfy

$$\mathrm{d}h_i = 0, \qquad \mathrm{d}b_i = 0, \qquad b = \star\mu h, \tag{5.24}$$

$$\mathrm{t}_{S_h} h = 0, \qquad \mathrm{t}_{S_b} b = 0, \tag{5.25}$$

and importantly on $S \subset M_i$

$$\mathrm{t}h_1 - \mathrm{t}h_2 = j_s \qquad \text{and } \mathrm{t}b_1 = \mathrm{t}b_2 \tag{5.26}$$

must hold. On the closed surface S we look for a pair of 1-forms $(e_s, j_s) \in \Omega^1(S) \times \Omega^1(S)$, the surface electric field and the surface current density, that satisfy

$$\mathrm{d}e_s = -\partial_t \mathrm{t}b_1 = -\partial_t \mathrm{t}b_2, \qquad \mathrm{d}j_s = 0, \qquad j_s = d\sigma \star_s e_s, \tag{5.27}$$

where σ is the physical conductivity in the 3-dimensional volume and d is the thickness of the skin depth region. The effective conductivity $d\sigma$ is justified by the dimensional reduction of the problem [54].

Weak formulation

Since the conducting region is two dimensional, we can use Ampere's law conforming formulation be able to express all the unknown fields with 0-forms. Specifically, we will approximate h_1, h_2 and j_s by

$$h_1 = \mathrm{d}\psi_1 = \sum_{\sigma_i^0 \in M_1 \setminus S_h} \psi_i^1 \mathrm{d}\mathsf{n}^i, \tag{5.28}$$

$$j_s = \partial_t \mathrm{d}\chi = \partial_t \sum_{\sigma_i^0 \in S} \chi_i \mathrm{d}\mathsf{n}^i, \tag{5.29}$$

where following [56], the extra time derivative is used to reach a symmetric linear system in the end. Equations (5.24) - (5.27) yield the following system of weak equations:

$$\int_{M_1} \star\mu \mathrm{d}\psi_1 \wedge \mathrm{d}\mathsf{n}^j = \int_S \mathrm{t}\star\mu \mathrm{d}\psi_2 \wedge \mathrm{t}\mathsf{n}^j \quad \forall\, \mathsf{n}^j \in W^0(M_1, S_h), \tag{5.30}$$

$$\int_{M_2} \star\mu \mathrm{d}\psi_2 \wedge \mathrm{d}\mathsf{n}^j = \int_S \mathrm{t}\star\mu \mathrm{d}\psi_1 \wedge \mathrm{t}\mathsf{n}^j \quad \forall\, \mathsf{n}^j \in W^0(M_2, S_h), \tag{5.31}$$

$$\int_S \star_s \frac{1}{d\sigma} \mathrm{d}\chi \wedge \mathrm{d}\mathsf{n}^j = -\int_S \mathrm{t}\star\mu \mathrm{d}\psi_1 \wedge \mathsf{n}^j = -\int_S \mathrm{t}\star\mu \mathrm{d}\psi_2 \wedge \mathsf{n}^j \quad \forall\, \mathsf{n}^j \in W^0(S), \tag{5.32}$$

$$\partial_t \int_S \star_s \frac{1}{d\sigma} \mathrm{d}\chi \wedge \mathrm{d}\mathsf{n}^j = \int_S \star_s \frac{1}{d\sigma} \mathrm{t}\mathrm{d}\psi_1 \wedge \mathrm{d}\mathsf{n}^j - \int_S \star_s \frac{1}{d\sigma} \mathrm{t}\mathrm{d}\psi_2 \wedge \mathrm{d}\mathsf{n}^j \quad \forall\, \mathsf{n}^j \in W^0(S), \tag{5.33}$$

from which terms involving the either the composition t$\star$ or the scalar potential χ can be eliminated. Plugging in the approximations, we end up with a system of ordinary differential equations with unknown vectors $\boldsymbol{\psi}_1$ and $\boldsymbol{\psi}_2$. They are related to $\boldsymbol{\chi}$ by $\dot{\boldsymbol{\chi}} = \boldsymbol{\psi}_1 - \boldsymbol{\psi}_2$. For the surface current density $j_s = \partial_t \mathrm{d}\chi = \mathrm{dt}\psi_1 - \mathrm{dt}\psi_2$ holds.

Implementation aspects

In essence, the implementation is to fill the system matrix with the integral expressions

$$\int_{M_{1,2}} \star\mu \mathrm{d}\mathsf{n}^i \wedge \mathrm{d}\mathsf{n}^j, \quad \mathsf{n}^i, \mathsf{n}^j \in W^0(M_{1,2}, S_h) \quad \text{and} \tag{5.34}$$

$$\int_S \star_s \frac{1}{d\sigma} \mathrm{d}\mathsf{n}^i \wedge \mathrm{d}\mathsf{n}^j, \quad \mathsf{n}^i, \mathsf{n}^j \in W^0(S). \tag{5.35}$$

That is, it is the same expression with the difference that the other concerns a 3-manifold with Cartesian metric and other a 2-manifold with a pullback of the Cartesian metric.

However, for the Riemannian manifold interface they are the same expression, since its objects are indifferent about the dimension and the metric. Therefore, using the Riemannian manifold interface, a single function is able to compute both expressions. The code is actually the same we presented in section 5.4.1. Compare that to the traditional approach in which two functions would have been needed, the other being more complicated because of the unusual metric tensor.

For post-processing, one can approximate the actual current density j in the conducting plate by

$$j = \frac{1}{d\sigma} \varsigma^* j_s \wedge \nu, \tag{5.36}$$

where $\nu \in \Omega^1(M_1)$ satisfies $\sqrt{\langle \nu, \nu \rangle} = 1$ and $\nu(\varsigma_*^{-1} v) = 0$ for all tangent vectors v of S. That is, ν is a unit normal 1-form to S.

In Fig. 5.10 we depict an example problem where a conducting, curved plate is floating in an uniform, time harmonic magnetic field.

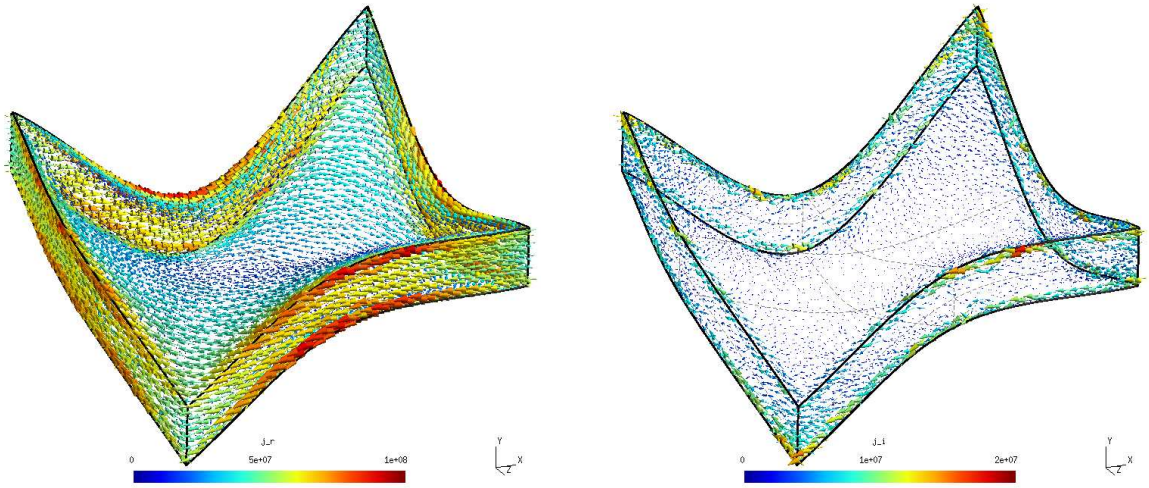


Figure 5.10: The approximate current density $j = 1/(d\sigma)\varsigma^* j_s \wedge \nu = j_r + ij_i$. The real part j_r is on the left and the imaginary part j_i is on the right.

Chapter 6

Conclusion

This study focused on the implementation of computational methods in the finite element method that exploit modern mathematical structures. In particular, we implemented homology and cohomology solver and a programming interface that separates the metric and the coordinates, and mimics coordinate-free objects on a Riemannian manifold. The former tool can be interfaced with finite element mesh generators and solvers, while the latter can be used to implement finite element pre- and post-processors and user interfaces for them.

We also showed that these kind of tools and the mathematical structures behind them are needed to bring computational electromagnetics to a firmer ground. They help to distinguish the general concepts and to avoid suffocating oneself with all the details involved.

6.1 Homology and cohomology solver

In the mathematical theory of boundary value problems on Riemannian manifolds, homology of the domain and cohomology class of the unknown field are intimately linked. Together they play a role in the existence and uniqueness of the solution to a boundary value problem. Therefore, they also need to be taken into account in the formulation and implementation of the numerical methods for boundary value problems. In this thesis we have shown that explicit computation of homology and cohomology is both practical and beneficial in the finite element modeling.

While our examples were mainly from the field of electromagnetics, the same principles apply also on other fields of physics and engineering, creating potential for future possibilities. Acknowledging this, the implemented homology and cohomology solver strives for efficient general, possibly relative, homology and cohomology computation. In contrast, for the 1-cohomology computation even more efficient methods might be developed with the loss of generality.

6.2 Riemannian manifold interface

The language of differential geometry is nowadays prominent in physics. However, engineering software tools still interface with the classical vector and tensor analysis which,

as we have argued, has some impractical traits. To demonstrate that a different approach might be viable, we have implemented a programming interface which mimics the framework of the differential geometry and the Riemannian manifold. The interface is designed to handle software engineering complexities arising from the free choice of the metric and the coordinate chart.

Such free choices unify many finite element modeling techniques, such as mesh deformation and "transformation techniques" used to model movement and infinite domains, for example. Furthermore, the properties of metamaterials that bend the paths of light rays, for example [53], can be modeled as the change of metric within the domain.

However, the freedom of choice of the metric and the coordinate chart increases the complexity of a computational software, as these choices need to be taken into account in the actual numerical computations. The solution is to use the metric and coordinate chart agnostic language of differential geometry as an interface, and underneath that interface the expressions are automatically translated to the metric and coordinate chart dependent computations. At the programming interface, the choice of the metric and the coordinate chart is controlled at one place.

6.3 Future developments

In the conclusions of [61] it was envisioned that homology computation should be brought closer to the engineering practice. The homology and cohomology solver presented in this thesis is a definite step towards that direction. We now have easy-to-use and efficient tool at our hands, and we have provided examples of its usage in engineering. In future, we hope homology and cohomology computation would be used by genuine working engineers.

The Riemannian manifold imitation presented in this thesis is an programming interface, not an user interface. Today, finite element modeling tools use classical vector analysis as their user interface. However in future, there might grow a demand for user interfaces where differential geometry plays the prominent part. If one used a Riemannian manifold programming interface similar to our implementation underneath, a Riemannian manifold user interface can be build on top of it with a minimal effort. Also, the the classical vector analysis approach is difficult in multiphysics environments due to field specific practices and tradition. As a new player on the field, differential geometry could be used as an unifying language. Therefore, one might construct classical field specific user interfaces that support engineer's intuition and experience, and couple those via a Riemannian manifold programming interface to build a maintainable multiphysics environment.

Bibliography

[1] P. R. Amestoy, I. S. Duff, J. Koster, and J.-Y. L'Excellent. A fully asynchronous multifrontal solver using distributed dynamic scheduling. *SIAM Journal on Matrix Analysis and Applications*, 23(1):15–41, 2001.

[2] D. N. Arnold, R. S. Falk, and R. Winther. Finite element exterior calculus, homological techniques, and applications. *Acta Numer.*, 15:1–155, 2006. doi:10.1017/S0962492906210018.

[3] D. N. Arnold, R. S. Falk, and R. Winther. Finite element exterior calculus: from hodge theory to numerical stability. *Bull. Amer. Math. Soc. (N.S.)*, 47:281–354, 2010. doi:10.1090/S0273-0979-10-01278-4.

[4] N. Bell and A. N. Hirani. Pydec: A Python library for Discrete Exterior Calculus. `http://code.google.com/p/pydec/`, 2008.

[5] N. Bell and A. N. Hirani. Pydec: Software and algorithms for discretization of exterior calculus. *ACM Transactions on Mathematical Software*, 2012. (to appear).

[6] A. Bossavit. Magnetostatic problems in multiply connected regions: Some properties of the curl operator. *Physical Science, Measurement and Instrumentation, Management and Education - Reviews, IEE Proceedings A*, 135(3):179–187, 1988. doi:10.1049/ip-a-1:19880028.

[7] A. Bossavit. Solving maxwell equations in a closed cavity, and the question of 'spurious modes'. *Magnetics, IEEE Transactions on*, 26(2):702–705, Mar 1990. doi:10.1109/20.106414.

[8] A. Bossavit. *Computational Electromagnetism*. Academic Press, 1998.

[9] A. Bossavit. How weak is the "weak solution" in finite element methods? *Magnetics, IEEE Transactions on*, 34(5):2429 –2432, sep 1998. doi:10.1109/20.717558.

[10] A. Bossavit. Computational electromagnetism and geometry. (5): The 'galerkin hodge'. *Journal of the Japan Society of Applied Electromagnetics*, 8(2):203 –209, 2000.

[11] G Carlsson. *Bull. Amer. Math. Soc.*, 46:255–308, 2009. doi:10.1090/S0273-0979-09-01249-X.

[12] CHomP Group. CHomP. `http://chomp.rutgers.edu/software/`, 2012.

[13] J. C. Crager and P.R. Kotiuga. Cuts for the magnetic scalar potential in knotted geometries and force-free magnetic fields. *Magnetics, IEEE Transactions on*, 38(2):1309–1312, Mar 2002. doi:10.1109/TMAG.2002.996334.

[14] G. F. D. Duff. Differential forms in manifolds with boundary. *Annals of Mathematics*, 56(1):pp. 115–127, 1952.

[15] P. Dular, F. Henrotte, F. Robert, A. Genon, and W. Legros. A generalized source magnetic field calculation method for inductors of any shape. *Magnetics, IEEE Transactions on*, 33(2):1398–1401, 1997. doi:10.1109/20.582518.

[16] P. Dular, P. Kuo-Peng, C. Geuzaine, N. Sadowski, and J.P.A. Bastos. Dual magnetodynamic formulations and their source fields associated with massive and stranded inductors. *Magnetics, IEEE Transactions on*, 36(4):1293 –1299, jul 2000. doi:10.1109/20.877677.

[17] P. Dular, W. Legros, and A. Nicolet. Coupling of local and global quantities in various finite element formulations and its application to electrostatics, magnetostatics and magnetodynamics. *Magnetics, IEEE Transactions on*, 34(5):3078 –3081, sep 1998. doi:10.1109/20.717720.

[18] P. Dłotko and R. Specogna. Efficient cohomology computation for electromagnetic modeling. *CMES: Computer Modeling in Engineering & Sciences*, 60(3):247–278, 2010. doi:doi:10.3970/cmes.2010.060.247.

[19] P. Dłotko and R Specogna. Cohomology in 3d magneto-quasistatics modeling. *Communications in Computational Physics*, 14(1):48–76, 2013. doi:doi:10.4208/cicp.151111.180712a.

[20] P. Dłotko and R. Specogna. A novel technique for cohomology computations in engineering practice. *Computer Methods in Applied Mechanics and Engineering*, 253(0):530 – 542, 2013. doi:10.1016/j.cma.2012.08.009.

[21] P. Dłotko, R. Specogna, and F. Trevisan. Automatic generation of cuts on large-sized meshes for the $t - \omega$ geometric eddy-current formulation. *Computer Methods in Applied Mechanics and Engineering*, 198(47–48):3765 – 3781, 2009. doi:10.1016/j.cma.2009.08.007.

[22] P. Dłotko, R. Specogna, and F. Trevisan. Voltage and current sources for massive conductors suitable with the geometric eddy-current formulation. *Magnetics, IEEE Transactions on*, 46(8):3069–3072, 2010. doi:10.1109/TMAG.2010.2043510.

[23] H. Flanders. *Differential Forms with Applications to the Physical Sciences*. Dover Publications, 1963.

[24] T. Frankel. *The Geometry of Physics*. Cambridge University Press, 1997.

[25] R. Geroch. *Mathematical Physics*. University of Chicago Press, 1985.

[26] C. Geuzaine and J.-F. Remacle. Gmsh: a Three-dimensional Finite Element Mesh Generator with Built-in Pre- and Post-processing Facilities. *International Journal for Numerical Methods in Engineering*, 79:1309–1331, 11 2009. doi:10.1002/nme.2579.

[27] V. Gol'dshtein, I. Mitrea, and M. Mitrea. Hodge decompositions with mixed boundary conditions and applications to partial differential equations on lipschitz manifolds. *Journal of Mathematical Sciences*, 172:347–400, 2011. doi:10.1007/s10958-010-0200-y.

[28] P. W. Gross and P. R. Kotiuga. *Electromagnetic Theory and Computation*. Cambridge University Press, 2004.

[29] V. Guillemin and A. Pollack. *Differential Topology*. AMS Chelsea Publishing Series. AMS Chelsea Pub., 2010.

[30] H. Si. Tetgen a Quality Tetrahedral Mesh Generator and Three-dimensional Delaunay Triangulator. `http://wias-berlin.de/software/tetgen/`, 2011.

[31] A. Hatcher. *Algebraic Topology*. Cambridge University Press, 2002.

[32] H. Helmholtz. Über integrale der hydrodynamischen gleichungen, welcher der wirbelbewegungen entsprechen. *Journal für die reine und angewandte Mathematik*, 55:25–55, 1858.

[33] F. Henrotte and K. Hameyer. An algorithm to construct the discrete cohomology basis functions required for magnetic scalar potential formulations without cuts. *Magnetics, IEEE Transactions on*, 39(3):1167–1170, 2003. doi:10.1109/TMAG.2003.810404.

[34] F. Henrotte, B. Meys, H. Hedia, P. Dular, and W. Legros. Finite element modelling with transformation techniques. *IEEE Transactions on Magnetics*, 35(3):1434–1437, 1999.

[35] M. Holst. Adaptive Numerical Treatment of Elliptic Systems on Manifolds. *Advances in Computational Mathematics*, 15:139–191, 2001.

[36] B. D. Saunders J.-G. Dumas, F. Heckenbach and V. Welker. In algebra, geometry and software systems. pages 177–206, March 2003.

[37] J.-G. Dumas, F. Heckenbach, B. D. Saunders and V. Welker. GAP Homology. `http://www.eecis.udel.edu/~dumas/Homology/`, 2011.

[38] T. Kaczynski, K. Mischaikow, and M. Mrozek. *Computational Homology*. Springer-Verlag New York Inc., 2004.

[39] T. Kaczynski, M. Mrozek, and M. Ślusarek. Homology Computation by Reduction of Chain Complexes. *Computers Math. Applic.*, 35(4):59 – 70, 1998.

[40] L. Kettunen. Fields and circuits in computational electromagnetism. *Magnetics, IEEE Transactions on*, 37(5):3393–3396, 2001. doi:10.1109/20.952621.

[41] L. Kettunen, K. Forsman, and A. Bossavit. Formulation of the Eddy Current Problem in Multiply Connected Regions in Terms of h. *Int. J. Numer. Meth. Engng.*, 41:935 – 954, 1998. doi:10.1002/(SICI)1097-0207(19980315)41:5<935::AID-NME321>3.0.CO;2-F.

[42] P. R. Kotiuga. *Hodge Decompositions and Computational Electromagnetics*. PhD thesis, Dept. Elect. Eng., McGill Univ., Montreal, Canada, 1984.

[43] P. R. Kotiuga. On making cuts for magnetic scalar potentials in multiply connected regions. *Journal of Applied Physics*, 61(8):3916–3918, 1987. doi:10.1063/1.338583.

[44] P. R. Kotiuga. Toward an algorithm to make cuts for magnetic scalar potentials in finite element meshes. *Journal of Applied Physics*, 63(8):3357–3359, 1988. doi:10.1063/1.340781.

[45] P. R. Kotiuga. An algorithm to make cuts for magnetic scalar potentials in tetrahedral meshes based on the finite element method. *Magnetics, IEEE Transactions on*, 25(5):4129–4131, 1989. doi:10.1109/20.42544.

[46] Rensselaer Polytechnic Institute. Image Processing Laboratory and D.J.R. Meagher. *Octree Encoding: a New Technique for the Representation, Manipulation and Display of Arbitrary 3-D Objects by Computer*. 1980.

[47] M. Mrozek and B. Batko. Coreduction homology algorithm. *Discrete & Computational Geometry*, 41:96–118, 2009. doi:10.1007/s00454-008-9073-y.

[48] J. R. Munkres. *Elements of Algebraic Topology*. Perseus Books, 1984.

[49] M. Pellikka, S. Suuriniemi, and L. Kettunen. Homology in electromagnetic boundary value problems. *Boundary Value Problems*, 2010(1):381953, 2010. doi:10.1155/2010/381953.

[50] M. Pellikka, S. Suuriniemi, and L. Kettunen. Powerful heuristics and basis selection bring computational homology to engineers. *Magnetics, IEEE Transactions on*, 47(5):1226 –1229, may 2011. doi:10.1109/TMAG.2010.2091109.

[51] M. Pellikka, S. Suuriniemi, L. Kettunen, and C. Geuzaine. Homology and cohomology computation in finite element modeling. *SIAM Journal on Scientific Computing*, 35(5):B1195–B1214, 2013. doi:10.1137/130906556.

[52] M. Pellikka, T. Tarhasaari, S. Suuriniemi, and L. Kettunen. A programming interface to the riemannian manifold in a finite element environment. *Journal of Computational and Applied Mathematics*, 246(0):225 – 233, 2013. doi:10.1016/j.cam.2012.10.022.

[53] J. B. Pendry, D. Schurig, and D. R. Smith. Controlling electromagnetic fields. *Science*, 312:1780–178, 2006. doi:10.1126/science.1125907.

[54] P. Raumonen. *Mathematical Structures for Dimensional Reduction and Equivalence Classification of Electromagnetic Boundary Value Problems*. PhD thesis, Dept. Elect. Eng., Tampere Univ. Technology, Tampere, Finland, 2009.

[55] Z. Ren. $t - \omega$ -formulation for eddy-current problems in multiply connected regions. *Magnetics, IEEE Transactions on*, 38(2):557 –560, mar 2002. doi:10.1109/20.996146.

[56] D. Rodger and N. Atkinson. Finite element method for 3d eddy current flow in thin conducting sheets. *Physical Science, Measurement and Instrumentation, Management and Education - Reviews, IEE Proceedings A*, 135(6):369–374, 1988.

[57] G. Schwarz. *Hodge Decomposition: a Method for Solving Boundary Value Problems*. Lecture notes in mathematics. Springer, 1995.

[58] J. Schöberl. Netgen, an Advancing Front 2d/3d-mesh Generator Based on Abstract Rules. *Comput. Visual. Sci.*, 1:41–52, 1997.

[59] H. J. S. Smith. *Philosophical Transactions of the Royal Society of London*, 151:293–326, 1861. doi:10.1098/rstl.1861.0016.

[60] R. Specogna. Complementary geometric formulations for electrostatics. *International Journal for Numerical Methods in Engineering*, 86(8):1041–1068, 2011.

[61] S. Suuriniemi. *Homological Computations in Electromagnetic Modeling*. PhD thesis, Dept. Elect. Eng., Tampere Univ. Technology, Tampere, Finland, 2004.

[62] A. Tausz, M. Vejdemo-Johansson, and H. Adams. Javaplex: A research software package for persistent (co)homology. `http://code.google.com/javaplex`, 2011.

[63] H. Whitney. *Geometric Integration Theory*. Princeton University Press, 1957.

[64] C. K. Yap. *Fundamental Problems of Algorithmic Algebra*. Oxford University Press, 2000.

Index

Tampereen teknillinen yliopisto
PL 527
33101 Tampere

Tampere University of Technology
P.O.B. 527
FI-33101 Tampere, Finland

ISBN 978-952-15-3279-5
ISSN 1459-2045